Маршрутизаторы серии RTT

R100, R200, R800

Справочник команд, версия ПО 1.37

2026

Содержание

1. ВВЕДЕНИЕ 5
1.1. Аннотация 5
1.2. Целевая аудитория 5
1.3. Условные обозначения 5
1.4. Примечания и предупреждения 5
1.5. Используемые сокращения 6
2. ПРАВИЛА ПОЛЬЗОВАНИЯ КОМАНДНОЙ СТРОКОЙ 8
3. СТРУКТУРА СИСТЕМЫ КОМАНД 13
3.1. Глобальный режим 13
3.2. Конфигурирование маршрутизатора 14
3.3. Типы и порядок именования интерфейсов маршрутизатора 26
3.4. Типы и порядок именования туннелей маршрутизатора 28
4. КОМАНДЫ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА 31
5. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ И КОНФИГУРАЦИЕЙ 60
6. НАСТРОЙКА TFTP-СЕРВЕРА 101
7. НАСТРОЙКА DNS 104
8. НАСТРОЙКА MDNS-REFLECTOR 108
9. НАСТРОЙКА ОБЩЕСИСТЕМНЫХ ПАРАМЕТРОВ 112
9.1. Общие параметры 112
9.2. Настройка подключения к серверу лицензирования 128
10. УПРАВЛЕНИЕ СИСТЕМНЫМИ ЧАСАМИ 135
11. НАСТРОЙКА AAA 155
12. КОНФИГУРИРОВАНИЕ И МОНИТОРИНГ ИНТЕРФЕЙСОВ 218
12.1. Общие команды 218
12.2. Физические интерфейсы 238
12.3. Агрегированные интерфейсы 249
12.4. E1/multilink 256
12.5. Беспроводные модемы 277
13. УПРАВЛЕНИЕ L2-ФУНКЦИЯМИ 291
13.1. Управление L2-функциями 291
13.2. Управление Spanning Tree 300
13.3. Настройка и мониторинг VLAN 324
13.4. Настройка и мониторинг Voice-VLAN 340
14. РАБОТА С АДРЕСНЫМИ ТАБЛИЦАМИ 345
15. НАСТРОЙКА VRF 361
16. НАСТРОЙКА IP-АДРЕСАЦИИ 366
17. НАСТРОЙКА IPV6-АДРЕСАЦИИ 374
18. УПРАВЛЕНИЕ ПРОФИЛЯМИ 391
19. УПРАВЛЕНИЕ NAT 414
20. УПРАВЛЕНИЕ ТУНЕЛИРОВАНИЕМ, VPN И УДАЛЕННЫМ ДОСТУПОМ 441
20.1. Управление ключами и сертификатами 441
20.1.1. Инфраструктура открытых ключей 441
20.1.2. Ручная генерация 464
20.2. Конфигурирование и мониторинг туннелей 481
20.3. Настройки IPsec VPN 547
20.3.1. Управление VPN. Настройки IKE 547
20.3.2. Управление VPN. Настройки IPsec 583
20.4. Управление VPN. Настройки удаленного доступа 603

20.4.1. Общие команды настройки удаленного доступа 603
20.4.2. Настройка L2TP over IPsec/PPTP-сервера 611
20.4.3. Настройка OpenVPN-сервера 622
20.4.4. Настройка WireGuard-сервера 637
21. МАРШРУТИЗАЦИЯ 642
21.1. Общие настройки маршрутизации 642
21.2. Общие команды анонсирования и приема маршрутов 658
21.3. Маршрутизация на основе политик (PBR) 672
21.4. Настройка связок ключей 708
21.5. Настройка параметров протокола BFD 712
21.6. Настройка статических маршрутов IPv4/IPv6 722
21.7. Настройка протокола BGP 730
21.8. Настройка протоколов RIP и RIPNG 771
21.9. Настройка протоколов OSPF и OSPFv3 787
21.10. Настройка протокола IS-IS 858
22. НАСТРОЙКИ MPLS 899
22.1. Общие настройки MPLS 899
22.2. Настройки протокола LDP 902
22.3. Настройки MPLS L2VPN 917
23. РЕЗЕРВИРОВАНИЕ 934
23.1. Управление VRRP 934
23.2. Управление IPv6 VRRP 949
23.3. Настройка MultiWAN 960
23.4. Настройка объектов отслеживания событий 993
23.5. Управление Dual-Homing 1003
23.6. Настройка failover-сервисов 1007
23.6.1. Настройка общих параметров для failover-сервисов 1007
23.6.2. Настройка резервирования DHCP 1011
23.6.3. Настройка резервирования Firewall/NAT 1015
23.6.4. Настройка Crypto-Sync 1023
24. УПРАВЛЕНИЕ КЛАСТЕРОМ 1027
25. БЕЗОПАСНОСТЬ 1034
25.1. Управление списками контроля доступа (ACL) 1034
25.2. Управление Firewall 1046
25.3. Управление логированием и защитой от сетевых атак 1090
25.3.1. Управление защитой от сетевых атак 1090
25.3.2. Управление оповещением о сетевых атаках 1106
25.4. Управление фильтрацией 1111
25.5. Управление системой предотвращения вторжений (IPS/IDS) 1121
25.5.1. Общие команды IPS/IDS 1121
25.5.2. Настройка политики IPS/IDS 1133
25.5.3. Настройка IPS 1138
25.5.4. Настройка параметров кэширования записей контентной фильтрации 1145
25.5.5. Настройка автообновления правил IPS/IDS, распространяемых по коммерческой лицензии.. 1146
25.5.6. Настройка автообновления правил IPS/IDS из внешних источников 1151
25.5.7. Настройка пользовательских правил IPS/IDS 1154
25.5.8. Настройка расширенных пользовательских правил 1181
26. УПРАВЛЕНИЕ QOS 1183
27. МОНИТОРИНГ И УПРАВЛЕНИЕ 1225

27.1. Управление Netflow 1225
27.2. Управление sFlow 1236
27.3. Управление SNMP 1242
27.4. Управление SYSLOG 1269
27.5. Настройка доступа SSH, Telnet 1293
27.6. Настройка протокола LLDP 1309
27.7. Настройка зеркалирования 1317
27.8. Настройка zabbix-agent и zabbix-proxy 1321
28. НАСТРОЙКА DHCP 1332
28.1. Управление DHCP-клиентом 1332
28.2. Управление IPv6 DHCP-клиентом 1342
28.3. Управление DHCP Relay агентом 1351
28.4. Управление IPv6 DHCP Relay агентом 1362
28.5. Настройка и мониторинг DHCP-сервера 1363
28.6. Настройка и мониторинг IPv6 DHCP-сервера 1379
29. НАСТРОЙКА SLA 1390
29.1. Общие команды настройки SLA 1390
29.2. Настройка Eltex-SLA 1397
30. НАСТРОЙКА WI-FI КОНТРОЛЛЕРА ТУННЕЛЕЙ 1427

1. ВВЕДЕНИЕ

1.1. Аннотация

В настоящем руководстве приведено описание команд CLI для администратора маршрутизаторов серии RTT R100, R200, R800 (в дальнейшем именуемого маршрутизатором или устройством).
Интерфейс командной строки (Command Line Interface, CLI) – интерфейс, предназначенный для управления, просмотра состояния и мониторинга устройства. Для работы потребуется любая установленная на ПК программа, поддерживающая работу по протоколам Telnet, SSH или прямое подключение через консольный порт (например, HyperTerminal).

1.2. Целевая аудитория

Справочник команд CLI предназначен для технического персонала, выполняющего настройку и мониторинг маршрутизатора серии RTT посредством интерфейса командной строки (CLI). Квалификация технического персонала предполагает знание основ работы стека протоколов TCP/IP, принципов построения Ethernet-сетей.

1.3. Условные обозначения

Обозначение	Описание
[ ]	В квадратных скобках в командной строке указываются необязательные параметры, но их ввод предоставляет определенные дополнительные опции.
{ }	В фигурных скобках в командной строке указываются возможные обязательные параметры. Необходимо выбрать один из параметров.
«,» **«-»**	Данные знаки в описании команды используются для указания диапазонов.
«|»	Данный знак в описании команды обозначает «или».
<Полужирный курсив>	Полужирным курсивом в угловых скобках указываются названия клавиш на клавиатуре.
`Текст в рамке`	В рамках с текстом указаны примеры и результаты выполнения команд.

1.4. Примечания и предупреждения

Примечания содержат важную информацию, советы или рекомендации по использованию и настройке устройства.

Предупреждения информируют пользователя о ситуациях, которые могут нанести вред программно-аппаратному комплексу, привести к некорректной работе системы или потере данных.

Информация содержит справочные данные об использовании устройства.

1.5. Используемые сокращения

- AAA – Authentication, Authorization, Accounting
- ARP – Address Resolution Protocol
- BGP – Border Gateway Protocol
- BRAS – Broadband Remote Access Server
- DHCP – Dynamic Host Configuration Protocol
- DNS – Domain Name System
- DDoS – Distributed Denial of Service
- DoS – Denial of Service
- FIFO – First In, First Out
- FTP – File Transfer Protocol
- FXO – Foreign eXchange Office
- FXS – Foreign eXchange Subscriber
- GRE – Generic Routing Encapsulation
- GRED – Gentle Random Early Detection
- HTTP – HyperText Transfer Protocol
- HTTPS – HyperText Transfer Protocol Secure
- ICMP – Internet Control Message Protocol
- IKE – Internet Key Exchange
- IDS – Intrusion Detection System
- IP – Internet Protocol
- IP4IP4 – IP in IP
- IPS – Intrusion Prevention System
- IPsec – IP Security
- L2TP – Layer 2 Tunneling Protocol
- L2TPv3 – Layer 2 Tunneling Protocol version 3
- LACP – Link Aggregation Control Protocol
- LAG – Link Aggregation Group
- LDAP – Lightweight Directory Access Protocol
- LLDP – Link Layer Discovery Protocol
- MAC – Media Access Control
- MTU – Maximum Transmission Unit
- mDNS – Multicast Domain Name System
- NAT – Network Address Translation
- NAS – Network Access Server
- NTP – Network Time Protocol
- OSPF – Open Shortest Path First
- PPP – Point-to-Point Protocol
- PPTP – Point-to-Point Tunneling Protocol
- PPPOE – Point-to-point protocol over Ethernet
- PSK – Pre-Shared Key
- QoS – Quality of Service
- RADIUS – Remote Authentication Dial In User Service
- RED – Random early detection
- RIP – Routing Informational Protocol

- SFTP – Secure Shell File Transfer Protocol
- SIP – Session Initiation Protocol
- SLA — Service Level Agreement
- SNMP – Simple Network Management Protocol
- SCP – Secure Copy Protocol
- SP – Strict Priority
- SSH – Secure Shell
- STP – Spanning Tree Protocol
- TACACS – Terminal Access Controller Access Control System
- TFTP – Trivial File Transfer Protocol
- URL – Uniform Resource Locator
- VLAN – Virtual Local Area Network
- VPN –Virtual Private Network
- VRF – Virtual Routing/Forwarding
- VRRP – Virtual Router Redundancy Protocol
- VTI – Virtual Tunnel Interface
- WAN – Wide Area Network
- WINS – Windows Internet Name Service
- WRR – Weighted Round Robin
- XAUTH – eXtended Authentication

2. ПРАВИЛА ПОЛЬЗОВАНИЯ КОМАНДНОЙ СТРОКОЙ

Для упрощения использования командной строки интерфейс поддерживает функцию автоматического дополнения команд. Эта функция активируется при неполно набранной команде и вводе символа табуляции <Tab>.

Другая функция, помогающая пользоваться командной строкой – контекстная подсказка. На любом этапе ввода команды можно получить подсказку о следующих элементах команды путем ввода вопросительного знака <?>.

Для упрощения команд всей системе команд придана иерархическая структура. Для перехода между уровнями иерархии предназначены специальные команды перехода. Это позволяет использовать менее объемные команды на каждом из уровней. Для обозначения текущего уровня, на котором находится пользователь, динамически изменяется строка приглашения системы.

Пример:

```
rtt> enable Включение 15 уровня привилегий
rtt# configure Переход в режим конфигурирования устройства
rtt(config)#
rtt(config)# exit возврат на уровень выше
rtt#
```

Для удобства использования командной строки реализована поддержка горячих клавиш, перечисленных в таблице 1.

Таблица 1 – Описание горячих клавиш командной строки CLI

Сочетание клавиши	Описание
Ctrl+D	Во вложенном командном режиме – выход в предыдущий командный режим (команда exit), в корневом командном режиме – выход из CLI (команда logout)
Ctrl+Z	Выход в корневой командный режим (команда top)
Ctrl+A	Переход в начало строки
Ctrl+E	Переход в конец строки
Ctrl+U	Удаление символов слева от курсора
Ctrl+K	Удаление символов справа от курсора
Ctrl+C	Очистка строки, а также обрыв выполнения команды
Ctrl+W	Удаление слова слева от курсора
Ctrl+B	Переход курсора на одну позицию назад
Ctrl+F	Переход курсора на одну позицию вперед

Сочетание клавиши	Описание
Ctrl+L	Очистка экрана

Для удобства чтения добавлен постраничный вывод большой по объему информации.

Пример:

```
rtt# show running-config
syslog max-files 3
syslog file-size 512
syslog file tmpsys:syslog/default
  severity info
exit
syslog console
  severity notice
exit

interface gigabitethernet 1/0/1
  ip address 10.100.14.1/24
exit
interface gigabitethernet 1/0/1.101
exit
interface gigabitethernet 1/0/2
  ip address 192.168.1.1/24
  ip address 10.100.100.2/24 secondary
exit
interface gigabitethernet 1/0/2.150
  ip address 10.150.150.2/24
exit
interface gigabitethernet 1/0/2.151
  ip address 10.151.151.15/24
exit
interface gigabitethernet 1/0/3
  ip address dhcp enable
exit
interface gigabitethernet 1/0/5.55
More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

Для отключения постраничного вывода в текущей сессии необходимо ввести команду:

```
rtt# terminal datadump
```

Интерфейс командной строки обеспечивает авторизацию пользователей и ограничивает доступ к командам на основании уровня привилегий, заданного администратором.

В системе может быть создано необходимое количество пользователей. Необходимый уровень привилегий задаётся индивидуально для каждого из них.

В заводской конфигурации в системе создан один пользователь с именем **admin** и паролем **password**.

Для обеспечения безопасности командного интерфейса команды распределены между 1, 10 и 15 уровнем привилегий:

- 1 уровень – доступен только мониторинг устройства;
- 10 уровень – доступно конфигурирование устройства, кроме создания пользователей, перезагрузки устройства, загрузки ПО;
- 15 уровень – нет ограничений.

Получение 15 уровня привилегий:

```
(rtt)> enable
(rtt)#
```

Возвращение на первоначальный уровень привилегий:

```
(rtt)# disable
(rtt)>
```

Система позволяет нескольким пользователям одновременно подключаться к устройству.

При вводе в качестве параметров строк, содержащих пробелы, вводимые данные можно заключить в кавычки или апострофы:

```
rtt(config)# interface gigabitethernet 1/0/1
rtt(config-if-gi)# description "[WAN] ISP-1 link"
rtt(config-if-gi)# exit
rtt(config)# interface gigabitethernet 1/0/2
rtt(config-if-gi)# description '[WAN] ISP-2 link'
rtt(config-if-gi)#
```

Основное отличие между кавычками и апострофами заключается в том, что при использовании кавычек часть спецсимволов внутри для корректного ввода необходимо экранировать, а текст между апострофами интерпретируется "как есть":

```
rtt(config)# banner login "first line\\nsecond line"
rtt(config)# banner exec 'first line\nsecond line'
rtt(config)#
```

Для уменьшения объема отображаемых данных в ответ на запросы пользователя и облегчения поиска необходимой информации можно воспользоваться фильтрацией. Для фильтрации информации требуется добавить в конец командной строки символ «|» и использовать одну из опций фильтрации:

- begin – выводить все после строки, содержащей заданный шаблон;
- count – выводить только количество строк, отображаемых в выводе команды (без вывода информации, отображаемой самой командой);
- counter – добавляет к выводимой информации номера строк;
- include – выводить все строки, содержащие заданный шаблон;
- exclude – выводить все строки, не содержащие заданный шаблон;
- until – выводить все до строки, содержащей заданный шаблон.

Шаблон поиска может быть задан регулярным выражением и содержать:

– Перечень символов. Можно определить перечень, заключив символы в квадратные скобки «[]». Соответствие будет проверяться по символам, перечисленным в перечне. Если первый символ перечня «^», то соответствие будет проверяться по любому символу, не входящему в перечень. Примеры:

- [-az] – 'a', 'z' и '-';
- [a-z] – все латинские буквы от 'a' до 'z'.

– Специальные символы и операторы:

- ^ – начало строки;
- $ – конец строки;
- . – любой символ в строке;
- * – ноль или более раз;
- | – или;
- + – одно или более повторений предшествующего элемента.

Вывод команды «show running-config syslog» без параметров:

```
rtt# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file tmpsys:syslog/default
  severity info
exit
syslog console
  severity notice
exit
```

Вывод команды «show running-config syslog» с параметром «begin»:

```
rtt# show running-config syslog | begin file-size
syslog file-size 512
syslog file tmpsys:syslog/default
  severity info
exit
syslog console
  severity notice
exit
```

Вывод команды «show running-config syslog» с параметром «include»:

```
rtt# show running-config syslog | include file-size
syslog file-size 512
```

Вывод команды «show running-config syslog» с параметром «exclude»:

```
rtt# show running-config syslog | exclude file-size
syslog max-files 3
syslog file tmpsys:syslog/default
  severity info
exit
syslog console
  severity notice
```

```
exit
```

Примеры использования регулярных выражений:

```
rtt# show interfaces status | include "^te.*"
te1/0/1          Up       Down     1500      a8:f9:4b:aa:05:d9
te1/0/2          Up       Down     1500      a8:f9:4b:aa:05:da
rtt# show interfaces status | include "^gi1/0/1[2568]"
gi1/0/12         Up       Down     1500      a8:f9:4b:aa:05:cc
gi1/0/15         Up       Down     1500      a8:f9:4b:aa:05:cf
gi1/0/16         Up       Down     1500      a8:f9:4b:aa:05:d0
gi1/0/18         Up       Down     1500      a8:f9:4b:aa:05:d2
rtt# show interfaces status | include "^[^tgI -]"
bridge 1         Up       Up       1500      a8:f9:4b:aa:05:c0
bridge 2         Up       Up       1500      a8:f9:4b:aa:05:c0
rtt# show interfaces status | include "(/3|/5)"
gi1/0/3          Up       Down     1500      a8:f9:4b:ab:0e:23
gi1/0/5          Up       Down     1500      a8:f9:4b:ab:0e:25
```

Для вывода информации, запрошенной командой show, в файл требуется добавить в конец командной строки соответствующий символ, путь и имя для создания или дополнения текстового файла.

Символы для вывода информации в файл:

- > – записать вывод команды в файл;
- >> – дописать вывод команды в конец файла.

Возможные пути для сохранения текстовых файлов вывода команд:

- flash:data/<FILE> – файл будет создан или дополнен в энергонезависимой памяти маршрутизатора в разделе flash:data;
- usb://<USB-NAME>:/<FILE> – файл будет создан или дополнен на USB-носителе с именем <USB-NAME>;
- mmc://<MMC-NAME>:/<FILE> – файл будет создан или дополнен на USB-носителе с именем <MMC-NAME>.

3. СТРУКТУРА СИСТЕМЫ КОМАНД

Система команд интерфейса командной строки маршрутизатора серии RTT разделена на иерархические уровни (разделы).

3.1. Глобальный режим

Верхний уровень иерархии команд приведен в таблице 2.

Таблица 2 – Иерархия командных режимов (верхний уровень)

Уровень	Команда входа	Вид строки подсказки	Команда выхода
Корневой режим (**ROOT**)		rtt> rtt#	exit end
Режим конфигурирования (**CONFIG**)	configure	rtt(config)#	
Режим отладки работы устройства (**DEBUG**)	debug	rtt(debug)#	

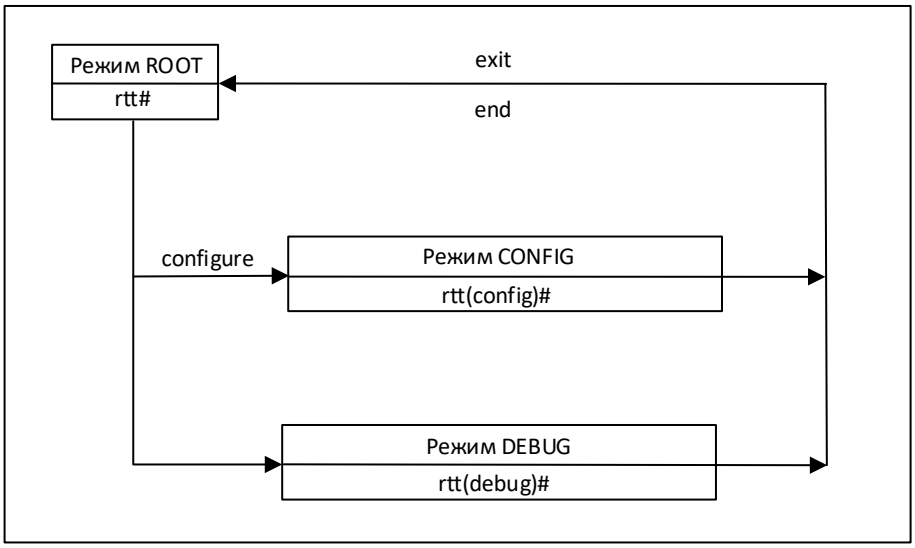

Рисунок 1 – Верхний уровень иерархии режимов команд

В корневом командном режиме (ROOT) осуществляется:

– работа с файлами конфигурации:

- применение;
- подтверждение;
- сброс;
- сохранение;
- отмена непримененных изменений;
- возврат к подтвержденной конфигурации.

– перезагрузка маршрутизатора;

– мониторинг работы и просмотр текущей конфигурации устройства.

Из корневого режима (ROOT) осуществляется переход к следующим разделам:

– режим конфигурирования устройства (CONFIG);

– режим отладки работы устройства (DEBUG).

3.2. Конфигурирование маршрутизатора

Конфигурирование маршрутизатора серии RTT выполняется в режиме **CONFIG**. Данный режим доступен из корневого режима (ROOT). Переход в режим конфигурирования осуществляется только в привилегированном режиме.

Для перехода из корневого режима (ROOT) необходимо выполнить следующие команды:

```
rtt> enable
rtt# configure
rtt(config)#
```

В режиме конфигурирования маршрутизатора серии RTT выполняется:

– управление системными часами;

– управление системным журналом;

– управление удаленным доступом;

– настройка QoS;

– настройка Spanning Tree;

– настройка VLAN;

– настройка статических маршрутов;

– настройка приоритетности протоколов маршрутизации;

– переход к режимам конфигурирования функций, описание режимов приведено в таблице 3.

Таблица 3 – Командные режимы для управления маршрутизатором

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка глобальных параметров **(CONFIG)**	**configure**	rtt(config)#	ROOT
Настройка GigabitEthernet-интерфейсов **(CONFIG-IF-GI)**	**interface gigabitethernet <PORT>**	rtt(config-if-gi)#	CONFIG
Настройка TengigabitEthernet-интерфейсов **(CONFIG-IF-TE)**	**interface tengigabitethernet <PORT>**	rtt(config-if-te)#	CONFIG
Настройка Twentyfivegigabitethernet-интерфейсов **(CONFIG-IF-TWE)**	**interface twentyfivegigabitethernet <PORT>**	rtt(config-if-twe)	CONFIG
Настройка FourtygigabitEthernet-интерфейсов **(CONFIG-IF-FO)**	**interface fourtygigabitethernet <PORT>**	rtt(config-if-fo)#	CONFIG
Настройка HundredgigabitEthernet-интерфейсов **(CONFIG-IF-HU)**	**interface hundredgigabitethernet <PORT>**	rtt(config-if-hu)#	CONFIG
Настройка OOB-интерфейса **(CONFIG-IF-OOB)**	**interface oob <PORT>**	rtt(config-if-oob)#	CONFIG
Настройка саб-интерфейсов **(CONFIG-IF-SUB)**	**interface gigabitethernet <PORT>.<VLAN> или interface tengigabitethernet <PORT>.<VLAN> или interface port-channel <CH>.<VLAN>**	rtt(config-if-sub)#	CONFIG
Настройка саб-интерфейсов **(CONFIG-IF-QINQ)**	**interface gigabitethernet <PORT>.<VLAN>.<VLAN> или interface tengigabitethernet <PORT>.<VLAN>.<VLAN> или interface port-channel <CH>.<VLAN>.<VLAN>**	rtt(config-if-qinq)#	CONFIG
Настройка виртуальных интерфейсов **(CONFIG-LOOPBACK)**	**interface loopback <PORT>**	rtt(config-loopback)#	CONFIG
Настройка E1-интерфейсов **(CONFIG-IF-E1)**	**interface e1 <PORT>**	rtt(config-if-e1)#	CONFIG
Настройка группы агрегации E1-каналов **(CONFIG-IF-MULTILINK)**	**interface multilink <PORT>**	rtt(config-if-multilink)#	CONFIG
Настройка FXO-портов **(CONFIG-VOICE-PORT-FXO)**	**interface voice-port <NUM>**	rtt(config-voice-port-fxo)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка FXS-портов **(CONFIG-VOICE-PORT-FXS)**	**interface voice-port <NUM>**	rtt(config-voice-port-fxs)#	CONFIG
Настройка виртуальных туннельных интерфейсов VTI **(CONFIG-VTI)**	**tunnel vti <VTI>**	rtt(config-vti)#	CONFIG
Настройка L2TPv3-туннелей **(CONFIG-L2TPV3)**	**tunnel l2tpv3 <L2TPV3>**	rtt(config-l2tpv3)#	CONFIG
Настройка GRE-туннелей **(CONFIG-GRE)**	**tunnel gre <GRE>**	rtt(config-gre)#	CONFIG
Настройка SoftGRE-туннелей **(CONFIG-SOFTGRE)**	**tunnel softgre <SOFTGRE>**	rtt(config-softgre)#	CONFIG
Настройка логических туннелей **(CONFIG-LT)**	**tunnel lt <LT>**	rtt(config-lt)#	CONFIG
Настройка саб-интерфейса на L2-туннеле **(CONFIG-SUBTUNNEL)**	**tunnel softgre <GRE>.<VLAN>**	rtt(config-subtunnel)#	CONFIG
Настройка IPv4-over-IPv4 туннелей **(CONFIG-IP4IP4)**	**tunnel ip4ip4 <IP4IP4>**	rtt(config-ip4ip4)#	CONFIG
Настройка WireGuard-туннелей **(CONFIG-WIREGUARD)**	**tunnel wireguard <WIREGUARD>**	rtt(config-wireguard)#	CONFIG
Настройка пира для WireGuard-туннеля **(CONFIG-WIREGUARD-PEER)**	**peer <PEER>**	rtt(config-wireguard-peer)#	CONFIG-WIREGUARD
Настройка сетевых мостов **(CONFIG-BRIDGE)**	**bridge <BRIDGE>**	rtt(config-bridge)#	CONFIG
Настройка VLAN **(CONFIG-VLAN)**	**vlan <VLAN>**	rtt(config-vlan)#	CONFIG
Настройка сетевой политики для определения VLAN по LLDPDU-сообщениям клиента **(CONFIG-NET-POLICY)**	**network-policy <NAME>**	rtt(config-net-policy)	CONFIG
Настройка пула адресов DHCP-сервера **(CONFIG-DHCP-SERVER)**	**ip dhcp-server pool <NAME>**	rtt(config-dhcp-server)#	CONFIG
Настройка DHCP опции 60 **(CONFIG-DHCP-VENDOR-ID)**	**ip dhcp-server vendor-class-id <NAME>**	rtt(config-dhcp-vendor-id)#	CONFIG
Настройка пула адресов DHCP-сервера **(CONFIG-IPV6-DHCP-SERVER)**	**ipv6 dhcp-server pool <NAME>**	rtt(config-ipv6-dhcp-server)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка DHCP опции 60 **(CONFIG-IPV6-DHCP-VENDOR-ID)**	**ipv6 dhcp-server vendor-class-id <NAME>**	rtt(config-dhcp-vendor-id)#	CONFIG
Настройка резервирования DHCP-сервера **(CONFIG-DHCP-SERVER-FAILOVER)**	**ip dhcp-server failover [vrf <VRF>]**	rtt(config-dhcp-server-failover)#	CONFIG
Настройка синхронизации крипто-файлов **(CONFIG-CRYPTO-SYNC)**	**crypto-sync**	rtt(config-crypto-sync)#	CONFIG
Настройка профиля приложений **(CONFIG-OBJECT-GROUP-APPLICATION)**	**object-group application <NAME>**	rtt(config-object-group-application)#	CONFIG
Настройка профиля IP-адресов **(CONFIG-OBJECT-GROUP-NETWORK)**	**object-group network <NAME>**	rtt(config-object-group-network)#	CONFIG
Настройка профиля связок IP-адресов и TCP/UDP-портов **(CONFIG-OBJECT-GROUP-ADDRESS-PORT)**	**object-group address-port <NAME>**	rtt(config-object-group-address-port)#	CONFIG
Настройка профиля TCP/UDP-портов **(CONFIG-OBJECT-GROUP-SERVICE)**	**object-group service <NAME>**	rtt(config-object-group-service)#	CONFIG
Настройка профиля URL **(CONFIG-OBJECT-GROUP-URL)**	**object-group url <NAME>**	rtt(config-object-group-url)#	CONFIG
Настройка поставщика категорий контентной фильтрации **(CONFIG-OBJECT-GROUP-CONTENT-FILTER)**	**object-group content-filter <NAME>**	rtt(config-object-group-content-filter)#	CONFIG
Настройка профиля доменных имен **(CONFIG-OBJECT-GROUP-DOMAIN-NAME)**	**object-group domain-name <NAME>**	rtt(config-object-group-domain-name)#	CONFIG
Настройка профиля категорий контентной фильтрации от вендора Kaspersky Laboratory **(CONFIG-OBJECT-GROUP-CF-KASPERSKY)**	**vendor <VENDOR>**	rtt(config-object-group-cf-kaspersky)#	CONFIG-OBJECT-GROUP-CONTENT-FILTER
Настройка профиля почтовых доменов и адресов почтовых ящиков **(CONFIG-OBJECT-GROUP-MAIL)**	**object-group email <NAME>**	rtt(config-object-group-email)#	CONFIG
Настройка профиля XAUTH **(CONFIG-ACCESS-PROFILE)**	**access profile <NAME>**	rtt(config-access-profile)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка пула IP-адресов **(CONFIG-POOL)**	**address-assignment pool <NAME>**	rtt(config-pool)#	CONFIG
Настройка шлюза протокола IKE **(CONFIG-IKE-GATEWAY)**	**security ike gateway <NAME>**	rtt(config-ike-gw)#	CONFIG
Настройка политики протокола IKE **(CONFIG-IKE-POLICY)**	**security ike policy <NAME>**	rtt(config-ike-policy)#	CONFIG
Настройка профиля протокола IKE **(CONFIG-IKE-PROPOSAL)**	**security ike proposal <NAME>**	rtt(config-ike-proposal)#	CONFIG
Настройка политики набора протоколов IPsec **(CONFIG-IPSEC-POLICY)**	**security ipsec policy <NAME>**	rtt(config-ipsec-policy)#	CONFIG
Настройка профиля набора протоколов IPsec **(CONFIG-IPSEC-PROPOSAL)**	**security ipsec proposal <NAME>**	rtt(config-ipsec-proposal)#	CONFIG
Настройка VPN на основе набора протоколов IPsec **(CONFIG-IPSEC-VPN)**	**security ipsec vpn <NAME>**	rtt(config-ipsec-vpn)#	CONFIG
Настройка списка контроля доступа **(CONFIG-ACL)**	**ip access-list extended <NAME>**	rtt(config-acl)#	CONFIG
Настройка правила для списка контроля доступа **(CONFIG-ACL-RULE)**	**rule <ORDER>**	rtt(config-acl-rule)#	CONFIG-ACL
Настройка зоны безопасности **(CONFIG-ZONE)**	**security zone <NAME>**	rtt(config-zone)#	CONFIG
Настройка группы правил для пар зон безопасности **(CONFIG-ZONE-PAIR)**	**security zone-pair <FROM> <TO>**	rtt(config-zone-pair)#	CONFIG
Настройка правила для пары зон безопасности **(CONFIG-ZONE-PAIR-RULE)**	**rule <ORDER>**	rtt(config-zone-rule)#	CONFIG-ZONE-PAIR
Настройка сервиса трансляции адресов получателя **(CONFIG-DNAT)**	**nat destination**	rtt(config-dnat)#	CONFIG
Настройка пула IP-адресов и TCP/UDP-портов для DNAT **(CONFIG-DNAT-POOL)**	**pool <NAME>**	rtt(config-dnat-pool)#	CONFIG-DNAT
Настройка группы правил для DNAT **(CONFIG-DNAT-RULESET)**	**ruleset <NAME>**	rtt(config-dnat-ruleset)#	CONFIG-DNAT
Настройка правила для DNAT **(CONFIG-DNAT-RULE)**	**rule <ORDER>**	rtt(config-dnat-rule)#	CONFIG-DNAT-RULESET

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка сервиса трансляции адресов отправителя **(CONFIG-SNAT)**	**nat source**	rtt(config-snat)#	CONFIG
Настройка пула IP-адресов и TCP/UDP-портов для SNAT **(CONFIG-SNAT-POOL)**	**pool <NAME>**	rtt(config-snat-pool)#	CONFIG-SNAT
Настройка группы правил для SNAT **(CONFIG-SNAT-RULESET)**	**ruleset <NAME>**	rtt(config-snat-ruleset)#	CONFIG-SNAT
Настройка правила для SNAT **(CONFIG-SNAT-RULE)**	**rule <ORDER>**	rtt(config-snat-rule)#	CONFIG-SNAT-RULESET
Настройка пользователей системы **(CONFIG-USER)**	**username <NAME>**	rtt(config-user)#	CONFIG
Настройка локальной консоли **(CONFIG-LINE-CONSOLE)**	**line console**	rtt(config-line-console)#	CONFIG
Настройка удаленной консоли **(CONFIG-LINE-TELNET)**	**line telnet**	rtt(config-line-telnet)#	CONFIG
Настройка защищенной удаленной консоли **(CONFIG-LINE-SSH)**	**line ssh**	rtt(config-line-ssh)#	CONFIG
Настройка TACACS-сервера **(CONFIG-TACACS-SERVER)**	**tacacs-server host <ADDR>**	rtt(config-tacacs-server)#	CONFIG
Настройка RADIUS-сервера **(CONFIG-RADIUS-SERVER)**	**radius-server host <ADDR>**	rtt(config-radius-server)#	CONFIG
Настройка профиля RADIUS-серверов **(CONFIG-RADIUS-SERVER-PROFILE)**	**aaa radius-profile <NAME>**	rtt(config-aaa-radius-profile)#	CONFIG
Настройка DAS-сервера **(CONFIG-DAS-SERVER)**	**das-server <NAME>**	rtt(config-das-server)#	CONFIG
Настройка профиля DAS-серверов **(CONFIG-DAS-SERVER-PROFILE)**	**aaa das-profile <NAME>**	rtt(config-aaa-das-profile)#	CONFIG
Настройка LDAP-сервера **(CONFIG-LDAP-SERVER)**	**ldap-server host <ADDR>**	rtt(config-ldap-server)#	CONFIG
Настройка сервера для удаленного доступа по протоколу PPTP **(CONFIG-PPTP-SERVER)**	**remote-access pptp <NAME>**	rtt(config-pptp-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу PPTP **(CONFIG-PPTP-USER)**	**username <NAME>**	rtt(config-pptp-user)#	CONFIG-PPTP

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка сервера для удаленного доступа по протоколу L2TP **(CONFIG-L2TP-SERVER)**	**remote-access l2tp <NAME>**	rtt(config-l2tp-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу L2TP **(CONFIG-L2TP-USER)**	**username <NAME>**	rtt(config-l2tp-user)#	CONFIG-L2TP-SERVER
Настройка сервера для удаленного доступа по протоколу OpenVPN **(CONFIG-OPENVPN-SERVER)**	**remote-access openvpn <NAME>**	rtt(config-openvpn-server)#	CONFIG
Настройка пользователей для удаленного доступа по протоколу L2TP **(CONFIG-OPENVPN-USER)**	**username <NAME>**	rtt(config-openvpn-user)#	CONFIG-OPENVPN-SERVER
Настройка сервера для удаленного доступа по протоколу WireGuard **(CONFIG-WIREGUARD-SERVER)**	**remote-access wireguard <NAME>**	rtt(config-wireguard-server)#	CONFIG
Настройка пира для удаленного доступа по протоколу WireGuard **(CONFIG-WIREGUARD-SERVER-PEER)**	**peer <PEER>**	rtt(config-wireguard-server-peer)#	CONFIG-WIREGUARD-SERVER
Настройка клиента удаленного доступа по протоколу PPTP **(CONFIG-PPTP)**	**tunnel pptp <PPTP-ID>**	rtt(config-pptp)#	CONFIG
Настройка клиента удаленного доступа по протоколу PPPOE **(CONFIG-PPPOE)**	**tunnel pppoe <PPPOE-ID>**	rtt(config-pppoe)#	CONFIG
Настройка клиента удаленного подключения по протоколу L2TP **(CONFIG-L2TP)**	**tunnel l2tp <L2TP-ID>**	rtt(config-l2tp)#	CONFIG
Настройка клиента удаленного подключения по протоколу OpenVPN **(CONFIG-OPENVPN)**	**tunnel openvpn <OPENVPN-ID>**	rtt(config-openvpn)#	CONFIG
Настройка SNMP-пользователя **(CONFIG-SNMP-USER)**	**snmp-server <NAME>**	rtt(config-snmp-user)#	CONFIG
Настройка NTP-сервера или пира **(CONFIG-NTP)**	**ntp server <ADDR>** **ntp peer <ADDR>**	rtt(config-ntp)#	CONFIG
Настройка BGP-процесса **(CONFIG-BGP)**	**router bgp <AS>**	rtt(config-bgp)#	CONFIG
Настройка BGP-процесса в VRF **(CONFIG-BGP-VRF)**	**vrf <VRF>**	rtt(config-bgp-vrf)#	CONFIG-BGP

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка IPv4/IPv6-адресации BGP-процесса **(CONFIG-BGP-FAMILY)**	**address-family { ipv4 | ipv6 } unicast**	rtt(config-bgp-af)#	CONFIG-BGP
Настройка IPv4/IPv6-адресации BGP-процесса в VRF **(CONFIG-BGP-VRF-FAMILY)**	**address-family { ipv4 | ipv6 } unicast**	rtt(config-bgp-vrf-af)#	CONFIG-BGP-VRF
Настройка группирования BGP-соседей **(CONFIG-BGP-GROUP)**	**peer-group <NAME>**	rtt(config-bgp-group)#	CONFIG-BGP
Настройка группирования BGP-соседей в VRF **(CONFIG-BGP-VRF-GROUP)**	**peer-group <NAME>**	rtt(config-bgp-vrf-group)#	CONFIG-BGP-VRF
Настройка соседа BGP-процесса **(CONFIG-BGP-NEIGHBOR)**	**neighbor { <ADDR> | <IPV6-ADDR> }**	rtt(config-bgp-neighbor)#	CONFIG-BGP
Настройка соседа BGP-процесса в VRF **(CONFIG-BGP-VRF-NEIGHBOR)**	**neighbor { <ADDR> | <IPV6-ADDR> }**	rtt(config-bgp-vrf-neighbor)#	CONFIG-BGP-VRF
Настройка IPv4-/IPv6-адресации BGP-соседа **(CONFIG-BGP-NEIGHBOR-FAMILY)**	**address-family { ipv4 | ipv6 | vpnv4 } unicast**	rtt(config-bgp-neighbor-af)#	CONFIG-BGP-NEIGHBOR
Настройка IPv4-/IPv6-адресации BGP-соседа в VRF **(CONFIG-BGP-NEIGHBOR-FAMILY)**	**address-family { ipv4 | ipv6 | vpnv4 } unicast**	rtt(config-bgp-neighbor-af-vrf)#	CONFIG-BGP-VRF-NEIGHBOR
Настройка списка подсетей **(CONFIG-PL)**	**ip prefix-list <NAME>**	rtt(config-pl)#	CONFIG
Настройка списка подсетей **(CONFIG-IPV6-PL)**	**ipv6 prefix-list <NAME>**	rtt(config-ipv6-pl)#	CONFIG
Настройка маршрутной карты **(CONFIG-ROUTE-MAP)**	**route-map <NAME>**	rtt(config-route-map)#	CONFIG
Настройка правила маршрутной карты **(CONFIG-ROUTE-MAP-RULE)**	**rule <ORDER>**	rtt(config-route-map-rule)#	CONFIG-ROUTE-MAP
Настройка RIP-протокола **(CONFIG-RIP)**	**router rip**	rtt(config-rip)#	CONFIG
Настройка RIPNG-протокола **(CONFIG-RIPNG)**	**ipv6 router rip**	rtt(config-ripng)#	CONFIG
Настройка IS-IS-процесса **(CONFIG-ISIS)**	**router isis <ID>**	rtt(config-isis)#	CONFIG
Настройка OSPF-процесса **(CONFIG-OSPF)**	**router ospf <ID>**	rtt(config-ospf)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка OSPF-области **(CONFIG-OSPF-AREA)**	**area <ID>**	rtt(config-ospf-area)#	CONFIG-OSPF
Настройка виртуального соединения OSPF **(CONFIG-OSPF-VLINK)**	**virtual-link <ID>**	rtt(config-ospf-vlink)#	CONFIG-OSPF-AREA
Настройка OSPFv3-процесса **(CONFIG-IPV6-OSPF)**	**ipv6 router ospf <ID>**	rtt(config-ipv6-ospf)#	CONFIG
Настройка OSPFv3-области **(CONFIG-IPV6-OSPF- AREA)**	**area <ID>**	rtt(config-ipv6-ospf-area)#	CONFIG-IPV6-OSPF
Настройка виртуального соединения OSPFv3 **(CONFIG-IPV6-OSPF-VLINK)**	**virtual-link <ID>**	rtt(config-ipv6-ospf-vlink)#	CONFIG-IPV6-OSPF-AREA
Настройка списка ключей **(CONFIG-KEYCHAIN)**	**key-chain <KEYCHAIN>**	rtt(config-keychain)#	CONFIG
Настройка ключа **(CONFIG-KEYCHAIN-KEY)**	**key <ID>**	rtt(config-keychain-key)#	CONFIG-KEYCHAIN
Настройка параметров MSTP **(CONFIG-MST)**	**spanning-tree mst configuration**	rtt(config-mst)#	CONFIG
Настройка правил WAN **(CONFIG-WAN-RULE)**	**wan load-balance rule <ID>**	rtt(config-wan-rule)#	CONFIG
Настройка правил WAN (IPv6) **(CONFIG-IPV6-WAN-RULE)**	**Ipv6 wan load-balance rule <ID>**	rtt(config-ipv6-wan-rule)#	CONFIG
Настройка target-листов **(CONFIG-TARGET-LIST)**	**wan load-balance target-list <NAME>**	rtt(config-target-list)#	CONFIG
Настройка target-листов (IPv6) **(CONFIG-IPV6-TARGET-LIST)**	**Ipv6 wan load-balance target-list <NAME>**	rtt(config-ipv6-target-list)#	CONFIG
Настройка target **(CONFIG-WAN-TARGET)**	**target <ID>**	rtt(config-wan-target)#	CONFIG-TARGET-LIST
Настройка target (IPv6) **(CONFIG-IPV6-WAN-TARGET)**	**target <ID>**	rtt(config-ipv6-wan-target)#	CONFIG-IPV6-TARGET-LIST
Настройка объектов отслеживания событий **(CONFIG-TRACK)**	**track <ID>**	rtt(config-track)#	CONFIG
Настройка Wi-Fi Controller **(CONFIG-WIRELESS)**	**wireless-controller**	rtt(config-wireless)#	CONFIG
Настройка VRF **(CONFIG-VRF)**	**ip vrf <NAME>**	rtt(config-vrf)#	CONFIG
Настройка политики QoS **(CONFIG-POLICY-MAP)**	**policy-map <NAME>**	rtt(config-policy-map)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка класса QoS **(CONFIG-CLASS-MAP)**	**class-map <NAME>**	rtt(config-class-map)#	CONFIG
Настройка класса внутри политики QoS **(CONFIG-POLICY-MAP-CLASS)**	**class <NAME>**	rtt(config-class-policy-map)#	CONFIG
Настройка PPP-пользователя для аутентификации удаленной стороны **(CONFIG-PPP-USER)**	**ppp chap username <NAME>**	rtt(config-ppp-user)#	CONFIG-IF-E1
	ppp chap username <NAME>		CONFIG-IF-MULTILINK
	user <NAME>		CONFIG-CELLULAR-PROFILE
	user <NAME>		CONFIG-ACCESS-PROFILE
	username <NAME>		CONFIG-L2TP-SERVER
	username <NAME>		CONFIG-PPTP-SERVER
Настройка параметров резервирования конфигурации **(CONFIG-ARCHIVE)**	**archive**	rtt(config-archive)#	CONFIG
Настройка сервера сбора статистики Netflow **(CONFIG-NETFLOW-HOST)**	**netflow collector <ADDR>**	rtt(config-netflow-host)#	CONFIG
Настройка сервера сбора статистики sFlow **(CONFIG-SFLOW-HOST)**	**sflow collector <ADDR>**	rtt(config-sflow-host)#	CONFIG
Настройка сервера получения уведомлений SNMP **(CONFIG-SNMP-HOST)**	**snmp-server host <ADDR>**	rtt(config-snmp-host)#	CONFIG
Изменение пароля после истечения срока действия **(CHANGE-EXPIRED-PASSWORD)**	-	rtt(change-expired-password)#	-
Настройка профиля фильтрации http-запросов **(CONFIG-PROFILE)**	**ip http profile <NAME>**	rtt(config-profile)	CONFIG
Настройка общих параметров MPLS **(CONFIG-MPLS)**	**mpls**	rtt(config-mpls)#	CONFIG
Настройка общих параметров LDP **(CONFIG-LDP)**	**ldp**	rtt(config-ldp)#	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка IPv4-адресации протокола LDP **(CONFIG-LDP-AF-IPV4)**	**address-family ipv4**	rtt(config-ldp-af-ipv4)#	CONFIG-LDP
Настройка LDP-unicast-соседства **(CONFIG-LDP-NEIGH)**	**neighbor <ADDR>**	rtt(config-ldp-neig)#	CONFIG-LDP
Настройка IPv4-адресации протокола LDP для отдельного ip-интерфейса **(CONFIG-LDP-AF-IPV4-IF)**	**interface <IF>**	rtt(config-ldp-af-ipv4-if)#	CONFIG-LDP-AF-IPV4
Создание L2-MPLS-туннелей **(CONFIG-L2VPN)**	**l2vpn**	rtt(config-l2vpn)#	CONFIG-MPLS
Настройка L2-MPLS-туннеля **(CONFIG-L2VPN-EOMPLS)**	**vpls <NAME>** **p2p <NAME>**	rtt(config-l2vpn-vpls)# rtt(config-l2vpn-p2p)#	CONFIG-L2VPN
Настройка PW-class (шаблона pseudo-wire) **(CONFIG-L2VPN-PW-CLASS)**	**pw-class <WORD>**	rtt(config-l2vpn-pw-class)#	CONFIG-L2VPN
Настройка параметров pseudo-wire L2-MPLS-туннеля **(CONFIG-L2VPN-PW)**	**pw <PW_ID> <LSR_ID> [<NEIGH_ADDR>]**	rtt(config-l2vpn-pw)#	CONFIG-L2VPN-EOMPLS
Настройка BGP auto-discovery and singnaling **(CONFIG-AUTODISCOVERY-BGP)**	**autodiscovery bgp**	rtt(config-bgp)#	CONFIG-L2VPN-EOMPLS
Настройка политики IPS/IDS **(CONFIG-IPS-POLICY)**	**security ips policy <NAME>**	rtt(config-ips-policy)#	CONFIG
Настройка поставщика правил IPS/IDS **(CONFIG-IPS-VENDOR)**	**vendor <VENDOR>**	rtt(config-ips-vendor)#	CONFIG-IPS-POLICY
Настройка категорий правил IPS/IDS **(CONFIG-IPS-VENDOR-CATEGORY)**	**category <CATEGORY>**	rtt(config-ips-vendor-category)#	CONFIG-IPS-VENDOR
Настройка сервиса IPS/IDS **(CONFIG-IPS)**	**security ips**	rtt(config-ips)#	CONFIG
Настройка источника обновления правил IPS/IDS, распространяемых по коммерческой лицензии **(CONFIG-CONTENT-PROVIDER)**	**content-provider**	rtt(config-content-provider)#	CONFIG
Настройка менеджера лицензий **(CONFIG-LICENCE-MANAGER)**	**licence-manager**	rtt(config-licence-manager)	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка автообновления правил IPS/IDS из внешних источников **(CONFIG-IPS-AUTO-UPGRADE)**	**auto-upgrade**	rtt(config-ips-auto-upgrade)#	CONFIG-IPS
Настройка пользовательского сервера обновлений правил IPS/IDS **(CONFIG-IPS-UPGRADE-USER-SERVER)**	**user-server <WORD>**	rtt(config-ips-upgrade-user-server)#	CONFIG-IPS-AUTO-UPGRADE
Настройка категорий пользовательских правил IPS/IDS **(CONFIG-IPS-CATEGORY)**	**security ips-category user-defined <CATEGORY_NAME>**	rtt(config-ips-category)#	CONFIG
Настройка пользовательских правил IPS/IDS **(CONFIG-IPS-CATEGORY-RULE)**	**rule <ORDER>**	rtt(config-ips-category-rule)#	CONFIG-IPS-CATEGORY
Настройка расширенных пользовательских правил IPS/IDS **(CONFIG-IPS-CATEGORY-RULE-ADVANCED)**	**rule-advanced <ORDER>**	rtt(config-ips-category-rule-advanced)#	CONFIG-IPS-CATEGORY
Настройка параметров работы zabbix-agent **(CONFIG-ZABBIX-AGENT)**	**zabbix-agent**	rtt(config-zabbix-agent)#	CONFIG
Настройка параметров работы zabbix-proxy **(CONFIG-ZABBIX-PROXY)**	**zabbix-proxy**	rtt(config-zabbix-proxy)#	CONFIG
Настройка почтового домена **CONFIG-MAILSERVER-DOMAIN**	**mailserver domain <NAME>**	rtt(config-mailserver-domain)#	CONFIG
Настройка почтового сервера **CONFIG-MAILSERVER**	**mailserver**	rtt(config-mailserver)#	CONFIG
Настройка Syslog-файла **CONFIG-SYSLOG-FILE**	**syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> | hdd://<HDD>:/<NAME> | mmc://<MMC>:/<NAME> | USB://<USB>:/<NAME>}**	(config-syslog-file)	CONFIG
Настройка удаленного Syslog-сервера **CONFIG-SYSLOG-HOST**	**syslog host <HOSTNAME>**	(config-syslog-host)	CONFIG
Настройка TFTP-сервера **CONFIG-TFTP-SERVER**	**ip tftp server vrf <VRF>**	(config-tftp-server)	CONFIG
Настройка Firewall failover **CONFIG-FIREWALL-FAILOVER**	**ip firewall failover**	(config-firewall-failover)	CONFIG
Настройка IP failover **CONFIG-IP-FAILOVER**	**ip failover**	(config-failover)	CONFIG

Уровень	Команда входа	Вид строки подсказки	Предыдущий уровень
Настройка Cluster **CONFIG-CLUSTER**	**cluster**	(config-cluster)	CONFIG
Настройка unit в Cluster **CONFIG-CLUSTER-UNIT**	**unit <ID>**	(config-cluster-unit)	CONFIG-CLUSTER
Настройка PKI-сервера **CONFIG-PKI-SERVER**	**crypto pki server**	(config-pki-server)	CONFIG
Настройка PKI-клиента **CONFIG-TRUSTPOINT**	**crypto pki trustpoint <NAME>**	(config-trustpoint)	CONFIG
Настройка отличительного имени сертификата PKI-сервера **CONFIG-PKI-SERVER-SUBJECT-NAME**	**subject-name**	(config-pki-server-subject-name)	CONFIG-PKI-SERVER
Настройка отличительного имени сертификата PKI-клиента **CONFIG-TRUSTPOINT-SUBJECT-NAME**	**subject-name**	(config-trustpoint-subject-name)	CONFIG-TRUSTPOINT
Настройка альтернативных имен PKI-клиента **CONFIG-TRUSTPOINT-SAN**	**subject-alt-name**	(config-trustpoint-san)	CONFIG-TRUSTPOINT

3.3. Типы и порядок именования интерфейсов маршрутизатора

При работе маршрутизатора используются сетевые интерфейсы различного типа и назначения. Система именования позволяет однозначно адресовать интерфейсы по их функциональному назначению и местоположению в системе. Далее в таблице приведен перечень типов интерфейсов.

Таблица 4 – Типы и порядок именования интерфейсов маршрутизатора

Тип интерфейса	Обозначение
Физические интерфейсы	Обозначение физического интерфейса включает в себя его тип и идентификатор. Идентификатор физических интерфейсов имеет вид **<UNIT>/<SLOT>/<PORT>**, где • **<UNIT>** – номер устройства в группе устройств [1..4], • **<SLOT>** – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули, • **<PORT>** – порядковый номер порта.
Порты 1 Гбит/с	**gigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **gigabitethernet 1/0/12** Допускается использовать сокращенное наименование, например gi1/0/12.
Порты 10 Гбит/с	**tengigabitethernet <UNIT>/<SLOT>/<PORT>** Пример обозначения: **tengigabitethernet 1/0/2** Допускается использовать сокращенное наименование, например te1/0/2.
Порт OOB (Out-of-band)	**oob <UNIT>/<SLOT>/<PORT>** Пример обозначения: **oob 1/0/1**

Тип интерфейса	Обозначение
Группы агрегации каналов	Обозначение группы агрегации каналов включает в себя его тип и идентификатор. Идентификатор port-channel-интерфейсов может иметь вид **{ <CHANNEL_ID> | <UNIT>/<CHANNEL_ID> }**, где • **<UNIT>** – номер устройства в группе устройств [1..4], • **<CHANNEL_ID>** – порядковый номер группы агрегации каналов [1..12] Примеры обозначений: **port-channel 6** **port-channel 1/6** Допускается использовать сокращенное наименование, например, po1 или po1/1.
Субинтерфейсы	Обозначение субинтерфейса образуется из обозначения базового интерфейса и идентификатора (VLAN) саб-интерфейса, разделенных точкой. Примеры обозначений: **gigabitethernet 1/0/12.100** **tengigabitethernet 1/0/2.123** **port-channel 1.6** **port-channel 1/6.6** Идентификатор саб-интерфейса может принимать значения [1..4094].
Q-in-Q интерфейсы	Обозначение Q-in-Q интерфейса образуется из обозначения базового интерфейса, идентификатора сервисного VLAN и идентификатора пользовательского VLAN, разделенных точкой. Примеры обозначений: **gigabitethernet 1/0/12.100.10** **tengigabitethernet 1/0/2.45.12** **port-channel 1.6.34** **port-channel 1/6.6.34** Идентификатор сервисного и пользовательского VLAN может принимать значения [1..4094].
E1-интерфейсы	Обозначение E1-интерфейса включает в себя его тип и идентификатор. Идентификатор E1-интерфейсов имеет вид **< UNIT>/< SLOT>/< STREAM>**, где • **<UNIT>** – номер устройства в группе устройств [1..4], • **<SLOT>** – номер E1-модуля в составе устройства [0..12], • **<STREAM>** – порядковый номер E1-потока [1..1]. Пример обозначения: **e1 1/0/1** **На маршрутизаторах R800 не поддерживается работа модулей ToPGATE-WAN-E1 с аппаратной версией (hardware revision) 812.**
Группы агрегации E1-каналов	Обозначение группы агрегации E1-каналов включает в себя его тип и порядковый номер интерфейса: **multilink <CHANNEL_ID>** Пример обозначения: **multilink 3**
Логические интерфейсы	Обозначение логического интерфейса является порядковым номером интерфейса: Примеры обозначений: **loopback 4** **bridge 60** **service-port 1**

Тип интерфейса	Обозначение
Последовательные интерфейсы	Обозначение последовательного интерфейса включает в себя его тип и идентификатор. Идентификатор serial интерфейсов и line-aux интерфейсов имеет вид **< UNIT>/< SLOT>/< PORT>**, где • **<UNIT>** – номер устройства в группе устройств [1..4], • **<SLOT>** – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули, • **<PORT>** – порядковый номер порта. Пример обозначения: **serial 1/0/1, line aux 1/0/1**
USB-модемы	Обозначение USB-модема включает в себя его тип и порядковый номер: **modem <MODEM-NUM>** Пример обозначения: **modem 1**
FXS/FXO-порты	Обозначение FXS/FXO-портов включает в себя его тип и порядковый номер: **interface voice-port <NUM>** Пример обозначения: **voice-port 1**

1. Количество интерфейсов каждого типа зависит от модели маршрутизатора.
2. Текущая версия ПО поддерживает кластеризацию устройств единой модели. Номер unit в группе устройств может принимать значение в диапазоне от 1 до 4.
3. Некоторые команды поддерживают одновременную работу с группой интерфейсов. Для указания группы интерфейсов может быть использовано перечисление через запятую или указание диапазона идентификаторов через дефис «-».

Примеры указания групп интерфейсов:

```
interface gigabitethernet 1/0/1, gigabitethernet 1/0/5
interface tengigabitethernet 1/0/1-2
interface fortygigabitethernet 1/0/1-2
interface gi1/0/1-3,gi1/0/7,te1/0/1,fo1/0/1
```

3.4. Типы и порядок именования туннелей маршрутизатора

При работе маршрутизатора используются сетевые туннели различного типа и назначения. Система именования позволяет однозначно адресовать туннели по их функциональному назначению. Далее в таблице приведен перечень типов туннелей.

Таблица 5 – Типы и порядок именования туннелей маршрутизатора

Тип туннеля	Обозначение
L2TP-туннель	Обозначение L2TP-туннеля состоит из обозначения типа и порядкового номера туннеля: **l2tp <L2TP_ID>** Пример обозначения: **l2tp 1**

Тип туннеля	Обозначение
L2TPv3-туннель	Обозначение L2TPv3-туннеля состоит из обозначения типа и порядкового номера туннеля: **l2tpv3 <L2TPV3_ID>** Пример обозначения: **l2tpv3 1**
GRE-туннель	Обозначение GRE-туннеля состоит из обозначения типа и порядкового номера туннеля: **gre <GRE_ID>** Пример обозначения: **gre 1**
GRE саб-туннель	Обозначение GRE саб-туннеля состоит из обозначения типа, порядкового номера туннеля и идентификатора VLAN для саб-туннеля: **gre <GRE_ID>.<VLAN_ID>** Пример обозначения: **gre 1.200**
SoftGRE-туннель	Обозначение SoftGRE-туннеля состоит из обозначения типа, порядкового номера туннеля и, опционально, VLAN ID виртуального интерфейса: **softgre <GRE_ID>[.<VLAN>]** Примеры обозначения: **softgre 1, softgre 1.10**
IPv4-over-IPv4-туннель	Обозначение IPv4-over-IPv4-туннеля состоит из обозначения типа и порядкового номера туннеля: **ip4ip4 <IPIP_ID>** Пример обозначения: **ip4ip4 1**
IPsec-туннель	Обозначение виртуального IPsec-туннеля состоит из обозначения типа и порядкового номера туннеля: **vti <VTI_ID>** Пример обозначения: **vti 1**
Логический туннель (туннель между VRF)	Обозначение логического туннеля состоит из обозначения типа и порядкового номера туннеля: **lt <LT_ID>** Пример обозначения: **lt 1**
PPPoE-туннель	Обозначение PPPoE-туннеля состоит из обозначения типа и порядкового номера туннеля: **pppoe <PPPoE_ID>** Пример обозначения: **pppoe 1**
PPTP-туннель	Обозначение PPPTP-туннеля состоит из обозначения типа и порядкового номера туннеля: **pptp <PPTP_ID>** Пример обозначения: **pptp 1**
OpenVPN-туннель	Обозначение OpenVPN-туннеля состоит из обозначения типа и порядкового номера туннеля: **openvpn <OPENVPN_ID>** Пример обозначения: **openvpn 1**

Тип туннеля	Обозначение
WireGuard-туннель	Обозначение WireGuard-туннеля состоит из обозначения типа и порядкового номера туннеля: **wireguard <WG_ID>** Пример обозначения: **wireguard 1**

Количество туннелей каждого типа зависит от модели и ПО маршрутизатора.

4. КОМАНДЫ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

alarm

Данной командой устанавливаются значения порогов аварий.

Синтаксис

```
alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-ALARMS> } <VALUE>
no alarm { facility <FACILITY-ALARMS> | memory <MEMORY-ALARMS> | process <PROCESS-ALARMS> } <VALUE>
```

Параметры

<FACILITY-ALARMS> – пороги аварий, связанных с окружением (скорость FAN, температура процессора и датчиков). Принимает следующие значения:

- fan-speed high – порог скорости вращения вентилятора, выраженный в процентах от максимальной скорости вращения (0..100), при переходе через который от меньших значений к большим, генерируется SNMP trap eltexEnvFanSpeedHigh (авария). Значение этого порога должно быть больше, чем alarm facility fan-speed low;
- fan-speed low – порог скорости вращения вентилятора, выраженный в процентах от максимальной скорости вращения (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvFanSpeedHighOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility fan-speed high;
- temperature cpu critical high – порог температуры процессора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от меньших значений к большим, генерируется SNMP trap eltexEnvTempCritical (авария). Значение этого порога должно быть самым большим из всех значений температурных порогов процессора;
- temperature cpu critical low – порог температуры процессора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvTempCriticalOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility temperature cpu critical high, но больше, чем alarm facility temperature cpu overheat high;
- temperature <sensor> overheat high – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от меньших значений к большим, генерируется SNMP trap eltexEnvTempOverheat (авария). Значение этого порога должно быть меньше, чем alarm facility temperature <sensor> critical low, но больше, чем alarm facility temperature <sensor> overheat low;
- temperature <sensor> overheat low – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..100), при переходе через который от больших значений к меньшим, генерируется SNMP trap eltexEnvTempOverheatOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm facility temperature <sensor> overheat high;
- temperature <sensor> supercooling high – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..12), при переходе через противоположное значение которого от меньших значений к большим, генерируется SNMP trap eltexEnvTempSupercoolingOk (нормализация аварии). Этот порог предназначен для задания отрицательных температур, и его значение должно быть меньше, чем alarm facility temperature <sensor> supercooling low;

- temperature <sensor> supercooling low – порог температуры сенсора, выраженный в градусах по шкале Цельсия (0..15), при переходе через противоположное значение которого от больших значений к меньшим, генерируется SNMP trap eltexEnvTempSupercooling (авария). Этот порог предназначен для задания отрицательных температур, и его значение должно быть больше, чем alarm facility temperature <sensor> supercooling high;
  Поле сенсор может принимать следующие значения для каждой модели маршрутизатора:
  - R100/200: cpu, board;
  - R800: cpu, board, switch, sfp.

<MEMORY-ALARMS> – пороги аварий, связанных с свободным пространством NAND и RAM. Принимает следующие значения:

- free low-watermark flash high – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryLowOk (нормализация аварии). Значение этого порога должно быть самым большим из всех значений порогов количества свободной flash-памяти;
- free low-watermark flash low – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений к меньшим генерируется SNMP trap eltexEnvMemoryLow (авария). Значение этого порога должно быть меньше, чем alarm memory free low-watermark flash high;
- free low-watermark ram high – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryLowOk (нормализация аварии). Значение этого порога должно быть самым большим из всех значений порогов количества свободной RAM-памяти;
- free low-watermark ram low – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryLow (авария). Значение этого порога должно быть меньше, чем alarm memory free low-watermark ram high;
- reserve critical flash high – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryCriticalLowOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm memory free low-watermark flash low, но больше, чем alarm memory reserve critical flash low;
- reserve critical flash low – порог количества свободной flash-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryCriticalLow (авария). Значение этого порога должно быть меньше, чем alarm memory reserve critical flash high;
- reserve critical ram high – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvMemoryCriticalLowOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm memory free low-watermark ram low, но больше, чем alarm memory reserve critical ram low;
- reserve critical ram low – порог количества свободной RAM-памяти, выраженной в процентах от максимального (0..100), при переходе через который от больших значений количества к меньшим генерируется SNMP trap eltexEnvMemoryCriticalLow (авария). Значение этого порога должно быть меньше, чем alarm memory reserve critical ram high.

<PROCESS-ALARMS> – пороги аварий, связанных с загрузкой процессора. Принимает следующие значения:

- cpu threshold falling – порог утилизации CPU, выраженной в процентах от максимальной (0..100), при переходе через который от больших значений к меньшим генерируется SNMP trap eltexEnvCpuLoadHighOk (нормализация аварии). Значение этого порога должно быть меньше, чем alarm process cpu threshold rising;
- cpu threshold rising – порог утилизации CPU, выраженной в процентах от максимальной (0..100), при переходе через который от меньших значений к большим генерируется SNMP trap eltexEnvCpuLoadHigh (авария). Значение этого порога должно быть больше, чем alarm process cpu threshold falling.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# alarm facility fan-speed high 80
```

alarm enable journal

Данной командой включается запись журнала аварий.

Синтаксис

```
[no] alarm enable journal [ bras [<ALARM_NAME>] | environment [<ALARM_NAME>] |
interfaces [<ALARM_NAME>] | snmp [<ALARM_NAME>] | wifi [<ALARM_NAME>] ]
```

Параметры

Group	<ALARM_NAME>
bras	sessions-number-high
environment	pwrin
environment	pwrin-insert
environment	hdd
environment	fan
environment	fan-speed-high
environment	memory-flash-critical-low
environment	memory-flash-low

Group	<ALARM_NAME>
environment	memory-ram-critical-low
environment	memory-ram-low
environment	cpu-load
environment	cpu-critical-temp
environment	cpu-overheat-temp
environment	cpu-supercooling-temp
environment	cpu-dp-critical-temp
environment	cpu-dp-overheat-temp
environment	cpu-dp-supercooling-temp
environment	cpu-mgmt-critical-temp
environment	cpu-mgmt-overheat-temp
environment	cpu-mgmt-supercooling-temp
environment	board-critical-temp
environment	board-overheat-temp
environment	board-supercooling-temp
environment	sfp-overheat-temp
environment	sfp-supercooling-temp
environment	switch-overheat-temp
environment	switch-supercooling-temp
environment	phy-overheat-temp
environment	phy-supercooling-temp
environment	coprocessor-critical-temp
environment	coprocessor-overheat-temp
environment	coprocessor-supercooling-temp
interfaces	rx-utilization-high
interfaces	tx-utilization-high
interfaces	number-high
ports	port-counters-errors
wifi	wifi-tunnels-number-in-bridge-high

Group	<ALARM_NAME>
snmp	linkdown
snmp	linkup

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# alarm enable journal snmp linkdown
```

clear alarms

Данной командой осуществляется очистка записей об авариях.

Синтаксис

```
clear alarms { all | inactive }
```

Параметры

all – удаляются записи об архивных и активных авариях;

inactive – удаляются записи только об архивных авариях.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear alarms inactive
```

clear dns records

Данной командой осуществляется очистка DNS-записей из кэша и негативного кэша.

Синтаксис

```
clear dns records [ all | domain <NAME> ]
```

Параметры

all – удаляются все кэшированные DNS-записи;

<NAME> – имя хоста, задаётся строкой до 253 символов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# clear dns records domain Rusteletech
cleared 1 records, 0 negative records
```

clear ssh host

Данной командой осуществляется сброс сохраненного SSH-ключа удаленного хоста.

Синтаксис

```
clear ssh host [ { <ADDR> | <IPV6-ADDR> | <HOST> } [ port <PORT>] ]
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOST> – доменное имя, задается задаётся строкой до 254 символа;

<PORT> – номер TCP-порта, принимает значения [1..65535];

Без указания параметров – осуществляется сброс всех сохраненных SSH-ключей.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear ssh host 192.168.1.1
```

clear storage-device

Данной командой запускается форматирование подключенного носителя.

Синтаксис

```
clear storage-device { usb-dev://<ID> | usb://<USB-NAME> | mmc } <NAME>
```

Параметры

<ID> – номер подключенного USB-носителя по порядку;

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды show storage-devices.

mmc – ключ для проверки установленного mmc-носителя;

<NAME> – новое имя раздела, которое будет назначено после форматирования.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear storage-device usb://BC1E-2E16:/ SOFT
Formatting will erase all data on this device
Do you really want to continue? (y/N): y
Device clear success.
```

configure

Данная команда позволяет перейти в режим глобального конфигурирования.

Синтаксис

```
configure
configure terminal
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# configure
rtt(config)#
```

do

Команда do позволяет выполнять команды корневого режима (ROOT) из любого другого режима командного интерфейса.

Синтаксис

```
do <command>
```

Параметры

<command> – команда корневого режима.

Необходимый уровень привилегий

1

Командный режим

Все режимы, кроме корневого.

Пример

```
rtt(config)# do show version
Boot version:
  1.24.0.1 (2024-08-01 13:40:59)
SW version:
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
HW version:
  1v7
```

end

Команда служит для возврата в корневой командный режим (ROOT).

Синтаксис

```
end
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

Все режимы, кроме корневого.

exit

Данная команда служит для возврата на уровень выше в иерархической системе командных режимов.

При выполнении данной команды в режиме ROOT завершается сеанс работы пользователя с интерфейсом командной строки CLI маршрутизатора.

Синтаксис

```
exit
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

Все режимы.

help

Данной командой на дисплей выводится информация о работе с командной строкой.

Синтаксис

```
help
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

Все режимы.

history size

Данной командой можно изменить максимальное число последних введенных команд пользователя, которые сохранятся в истории команд текущей сессии. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
history size <SIZE>
no history size
```

Параметры

<SIZE> – число последних введенных команд, принимает значения [10..1000].

Значение по умолчанию

50

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# history size 20
```

logout

Данной командой завершается сеанс работы пользователя с интерфейсом командной строки CLI.

Синтаксис

```
logout
```

Параметры

Команда не содержит параметров

Необходимый уровень привилегий

1

Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

Пример

```
rtt# logout
```

lookup

Данная команда используется для обращения к DNS системе.

Синтаксис

```
lookup [ vrf <VRF>]  <NAME> [ detailed ] [ nameserver <IP> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<NAME> – имя хоста, задаётся строкой до 253 символов;

<IP> – IP-адрес хоста, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

detailed – информация выдаётся в детализированном формате.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# lookup Rusteletech.ru
Server:         127.0.0.1
Address:        127.0.0.1#53

Non-authoritative answer:
Name:   Rusteletech.ru
Address: 62.109.9.104
```

monitor

Данной командой включается мониторинг трафика на сетевом интерфейсе в режиме реального времени попакетно.

Реализована запись дампа трафика в файл (.pcap, .pcapng, .txt, ...) с последующей возможностью копирования на usb/mms/flash:data/tftp-server. Записанный в файл дамп трафика в раздел файловой системы flash:data/ лимитирован 1000 пакетами.

Синтаксис

```
monitor { <IF> | <TUN> } [ protocol <TYPE> [ source-port <SRC-PORT> ] [
destination-port <DST-PORT> ] [ port <PORT> ] ] [ source-address {<SRC-ADDR> |
<SRC-IPV6-ADDR> } ] [ destination-address { <DST-ADDR> | <DST-IPV6-ADDR> } ]
[address { <ADDR> | <IPV6-ADDR> } ] [ packets <VALUE> ] [ detailed ] [ file <META-
PATH> ] [ detailed ]
```

Параметры

<IF> – интерфейс или группа интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<SRC-ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<DST-ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DST-IPV6-ADDR> – IPv6-адрес получателя, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<ADDR> – IP-адрес отправителя или получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес отправителя или получателя, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<SRC-MAC> – MAC-адрес отправителя ethernet-кадра, задаётся в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<DST-MAC> – MAC-адрес получателя ethernet-кадра, задаётся в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<SRC-MAC> – MAC-адрес отправителя или получателя ethernet-кадра, задаётся в виде HH:HH:HH:HH:HH:HH, где каждая HH-часть принимает значения в шестнадцатеричном формате [0..FF];

<TYPE> – тип протокола, принимает значения: tcp, udp, icmp, icmp6, igrp, igmp, arp, gre, ipip, esp, ah, eigrp, ospf, pim, vrrp, l2tp, RDP или номер протокола [0..255]**;**

<SRC-PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535];

<DST-PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];

<PORT> – номер TCP/UDP-порта отправителя или получателя, принимает значения [1..65535];

<VALUE> – количество пакетов, после получения которых анализ будет остановлен, указывается в диапазоне [1…4294967295];

detailed – информация выдаётся в детализированном формате.

<FILE> – запись дампа трафика в файл с любым расширением – .pcap, .pcapng, .txt и др.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# monitor gigabitethernet 1/0/5 detailed
23:37:44.324049 d8:50:e6:d2:f0:46 > a8:f9:4b:aa:03:a5, ethertype IPv4 (0x0800),
length 98: (tos 0x0, ttl
 64, id 50760, offset 0, flags [DF], proto ICMP (1), length 84)
    10.255.100.1 > 10.255.100.5: ICMP echo request, id 11730, seq 19, length 64
rtt# monitor gigabitethernet 1/0/5 protocol tcp file flash:data://dump.pcap
rtt# copy flash:data://dump.pcap tftp://<ip-address>:<port>:/copied_dump.pcap
```

ping

Данная команда используется для проверки доступности указанного сетевого устройства.

Синтаксис

```
ping [ vrf <VRF>] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> [ { ip | ipv6 } ] [ ttl <TTL>
] [ packets <COUNT> | unlimited ] [ size <SIZE> ] [ timeout <TIMEOUT> ] [ interval
<INTERVAL> ] [ source { ip { <SRC-ADDR> | <SRC-IPV6-ADDR> } | interface <IF> |
tunnel <TUN> } ] [ data <HEX> ] [ dscp <DSCP> ] [ flood ] [ detailed ] [ strategy
<STRATEGY> ] [ nodeinfo <INFO> ] [ broadcast ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

- ip – проверять доступность при помощи ipv4-пакетов;
- ipv6 – проверять доступность при помощи ipv6-пакетов.

<TTL> – время жизни IP-пакета, принимает значение [1..255], по умолчанию 64;

<COUNT> – количество передаваемых пакетов [1..4294967295], по умолчанию 5;

<SIZE> – размер icmp-payload в байтах, принимает значение [1..65468], по умолчанию 56 байт, что соответствует 64 байтам после добавления заголовка ICMP и 84 байтам после добавления ip-заголовка;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждет два RTTs. Принимает значение [1..60], по умолчанию 1 секунда;

<INTERVAL> – интервал между отправками icmp-пакетов в миллисекундах, принимает значение [200..60000], по умолчанию 1000.

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<HEX> – шаблон данных, которым будет заполняться пакет, задаётся числом в шестнадцатеричной системе до 16 байт;

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

flood – при указании данной команды пакеты будут отправляться с максимальной скоростью, ответы от устройства не отображаются до окончания выполнения команды;

broadcast – при указании данной команды будет разрешено отправлять пакеты на широковещательный адрес;

detailed – при указании данного ключа будет выводиться полная информация о icmp-ответах и общая статистика работы команды. Без указания – только статистика.

<STRATEGY> – стратегия фрагментации пакетов, принимает одно из следующих значений:

- allow-fragmentation – разрешить фрагментацию, не устанавливать флаг DF (don't fragment);
- discovery-pmtu – выполнять изучение PMTU (Path MTU), фрагментировать локально, если размер пакета слишком большой;
- disallow-fragmentation – запретить фрагментацию, в том числе локальную.

<INFO> – только для IPv6. Отправка ICMPv6 Node Information Queries (RFC4620), вместо Echo Request принимает одно из следующих значений:

- name – запрос DNS-имен(и) узла;
- ipv6 – запрос IPv6-адресов узла;
- ipv6-global – запрос глобальных IPv6-адресов узла;
- ipv6-sitelocal – запрос site-local IPv6-адресов узла;
- ipv6-linklocal – запрос link-local IPv6-адресов узла;
- ipv6-all – запрос unicast IPv6-адресов узла;
- ipv4 – запрос IP-адресов узла;
- ipv4-all – запрос IP-адресов со всех сетевых интерфейсов узла.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# ping 192.168.100.39 packets 5 size 1400 detailed
PING 192.168.100.39 (192.168.100.39) 1400(1428) bytes of data.
1408 bytes from 192.168.100.39: icmp_req=1 ttl=64 time=0.084 ms
1408 bytes from 192.168.100.39: icmp_req=2 ttl=64 time=0.053 ms
1408 bytes from 192.168.100.39: icmp_req=3 ttl=64 time=0.082 ms
1408 bytes from 192.168.100.39: icmp_req=4 ttl=64 time=0.051 ms
1408 bytes from 192.168.100.39: icmp_req=5 ttl=64 time=0.075 ms
--- 192.168.100.39 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.051/0.069/0.084/0.014 ms
rtt# ping ipv6 fc00::1
PING fc00::1(fc00::1) 56 data bytes
64 bytes from fc00::1: icmp_seq=1 ttl=64 time=0.379 ms
64 bytes from fc00::1: icmp_seq=2 ttl=64 time=0.161 ms
--- fc00::1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 0.161/0.270/0.379/0.109 ms
```

reload cancel

Данной командой осуществляется отмена ранее запланированной перезагрузки устройства.

Синтаксис

```
reload cancel
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# reload cancel
```

reload system

Данной командой осуществляется перезагрузка устройства.

Синтаксис

```
reload system [ with-configuration <FILE> ] { in { <MINUTES> | <HOUR>:<MINUTES>} |
at <TIME> <DATE> }
```

Параметры

<FILE> – путь до файла конфигурации, применяемого после перезагрузки. Может принимать следующие значения:

- flash:backup/FILE
- flash:data/FILE

<HOUR> – количество часов до перезагрузки,

<MINUTES> – количество минут до перезагрузки,

<TIME> – точное время перезагрузки в формате HH:MM:SS:

HH – часы в диапазоне [0..23],

MM – минуты в диапазоне [0..59],

SS – секунды в диапазоне [0..59].

<DATE> – дата перезагрузки в формате DAY MONTH YEAR:

DAY – день перезагрузки, принимает значение в диапазоне [1..31],

MONTH – месяц перезагрузки, принимает одно из значений:

- january
- february
- march
- april
- may
- june
- july
- august
- september
- october
- november
- december

YEAR – год перезагрузки, принимает значение в диапазоне [2001..2037].

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# reload system
```

show alarms brief

Данной командой выводится краткая история аварий на маршрутизаторе (активных и завершенных).

Синтаксис

```
show alarms brief
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show alarms brief

   History Alarms
   ~~~~~~~~~~~~~~
Severity  Group       Set time            Clear time          Description
--------   --------   -------------        --------------     ---------------------
major   environment 2000-03-31 17:27:38 2000-03-31 17:31:53 Reserve Power Supply Fault
```

show alarms brief active

Данной командой выводится краткая информация о текущих (активных) авариях на маршрутизаторе.

Синтаксис

```
show alarms brief active
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show alarms brief active
   History Alarms
   ~~~~~~~~~~~~~~
Severity   Group         Set time            Clear time       Description
--------   ---------   -----------------     ----------   ----------------------------------
-
notify     environment   2000-03-31 16:47:05   -          Reserve Power Supply removed
```

show alarm settings

Данной командой выводится информация о настройках порогов срабатывания аварий.

Синтаксис

```
show alarm settings <TYPE>
```

Параметры

<TYPE> – тип аварий:

- facility – пороги аварий, связанных с окружением (скорость FAN, температура процессора и датчиков);
- memory – пороги аварий, связанных с свободным пространством NAND и RAM;

- process – пороги аварий, связанных с загрузкой процессора.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show alarm facility
    fan-speed:
        high:                   80
        low:                    75
    temperature CPU:
        critical high:          95
        critical low:           92
        overheat high:          80
        overheat low:           78
        supercooling high:      -12
        supercooling low:       -15
    temperature sensor1:
        overheat high:          60
        overheat low:           57
        supercooling high:      -12
        supercooling low:       -15
    temperature sensor2:
        overheat high:          60
        overheat low:           57
        supercooling high:      -12
        supercooling low:       -15
    temperature sensor3:
        overheat high:          60
        overheat low:           57
        supercooling high:      -12
        supercooling low:       -15
```

show history

Данной командой на дисплей выводится информация о командах, которые использовались в текущей сессии, или о количестве сохраняемых команд.

Синтаксис

```
show history [ size | <NUM> ]
```

Параметры

size – максимальное число последних введенных команд пользователя, которые сохранятся в истории команд текущей сессии.

<NUM> – количество последних введенных команд пользователя, которые необходимо отобразить [0..100].

Значение по умолчанию

0. Соответствует выводу всей истории введенных команд.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show history
    1  enable
    2  show history
    3  configure
    4  service nat
    5  service nat source
    6  exit
    7  show history
```

show system reload

Данной командой на дисплей выводится информация о командах, дате и времени запланированной перезагрузки.

Синтаксис

```
show system reload
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show system reload
Schedule:             System reload is scheduled for 2025-08-10 04:00:00
```

```
Last reload reason:   System reload reason is unknown
```

show tech-support

Данная команда используется для сбора комплексной системной информации с устройства.

Синтаксис

```
show tech-support
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример:

```
rtt# show tech-support
|****************************************| 100% Success.
Show tech-support output available at: flash:data/20110413_112242_show_tech-supz

The created archive may contain private data. You can delete them yourself
Execute time 58.995103 sec
```

ssh

Данная команда используется для подключения к удаленному узлу по протоколу SSH.

Синтаксис

```
ssh [ vrf <VRF>] <USERNAME> { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ] [
version <VERSION> ] [ source { <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<USERNAME> – имя пользователя, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 253 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535]. По умолчанию установлено 22;

<VERSION> – версия SSH-протокола, принимает значения [1..2]. По умолчанию используется версия 1;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# ssh tester 10.100.100.1
The authenticity of host '10.100.100.1 (10.100.100.1)' can't be established.
ECDSA key fingerprint is db:e4:0a:93:59:87:7d:9f:90:5c:19:a3:e7:97:ec:d5.
Are you sure you want to continue connecting (yes/no)? yes
%AAA-I-SSH: Warning: Permanently added '10.100.100.1' (ECDSA) to the list of known
hosts.
tester@10.100.100.1's password:
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
  System information as of Mon May 25 09:25:10 NOVT 2015
Last login: Tue May 12 19:39:11 2015
(tester@kubuntu  ~) $
```

telnet

Данная команда используется для подключения к удаленному узлу по протоколу Telnet.

Синтаксис

```
telnet [ vrf <VRF>] { <ADDR> | <IPV6-ADDR> | <HOSTNAME> } [ port <PORT> ] [ source
{ <SRC-ADDR> | <SRC-IPV6-ADDR> } ] [ dscp <DSCP> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 253 символов;

<PORT> – номер TCP-порта, прослушиваемого SSH-сервером, принимает значения [1..65535], по умолчанию 23;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# telnet 10.100.100.1
Entering character mode
Escape character is '^]'.
Ubuntu 14.04.2 LTS
kubuntu login: tester
Password:
Last login: Mon May 25 15:23:06 NOVT 2015 from sw31-1.Rusteletech.loc on pts/16
Welcome to Ubuntu 14.04.2 LTS (GNU/Linux 3.13.0-51-generic x86_64)
 * Documentation:  https://help.ubuntu.com/
  System information as of Mon May 25 15:23:01 NOVT 2015
(tester@kubuntu  ~) $
```

terminal datadump

Команда используется для выключения постраничного режима вывода трассировок для текущей сессии.

Использование отрицательной команды включает постраничный режим вывода трассировок.

Синтаксис

```
[no] terminal datadump
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# terminal datadump
```

terminal mode

Команда используется для переключения режима работы терминала для текущей сессии.

Использование отрицательной команды включает человекочитаемый режим работы терминала.

Синтаксис

```
terminal mode { human | machine }
```

Параметры

human – человекочитаемый режим работы терминала, содержимое терминала перерисовывается при каждом пользовательском вводе, предназначен для интерактивного использования;

machine – машинный режим работы терминала, содержимое терминала не перерисовывается при пользовательском вводе, предназначен для взаимодействия с CLI через скрипты.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# terminal mode machine
```

terminal resize

Команда используется для масштабирования размера терминала под размер окна при использовании консольного подключения.

Синтаксис

```
terminal resize
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# terminal resize
```

traceroute

Данная команда используется для трассировки маршрута до указанного сетевого устройства.

Синтаксис

```
traceroute [ vrf <VRF> ] { <ADDR> | | <IPV6-ADDR> | <HOSTNAME> [ { ip | ipv6 } ] }
[ first-ttl <FIRST-TTL> ] [ max-ttl <MAX-TTL> ] [ timeout <TIMEOUT> ] [ source { ip
{ <SRC-ADDR> | <SRC-IPV6-ADDR> } | interface <IF> | tunnel <TUN> } ] [ dscp <DSCP>
] [ protocol { icmp | udp [ <PORT> ] | tcp [ <PORT> ] } ] [ gateway { <GW-ADDR> |
<GW-IPV6-ADDR> } ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа;

<ADDR> – IP-адрес устройства, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес устройства, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<HOSTNAME> – DNS-имя устройства, задаётся строкой до 255 символов;

- ip – проводить трассировку при помощи ipv4-пакетов;

- ipv6 – проводить трассировку при помощи ipv6-пакетов.

<FIRST-TTL> – время жизни IP-пакета, значение с которого начинается трассировка маршрута, принимает значение [1..255], по умолчанию 1;

<MAX-TTL> – время жизни IP-пакета, значение на котором заканчивается трассировка маршрута, принимает значение [1..255], по умолчанию 30;

<TIMEOUT> – время ожидания ответа, в секундах. Опция влияет на таймаут, если отсутствуют какие-либо ответы, в противном случае утилита ждет два RTTs. Принимает значение [1..60], по умолчанию 5 секунд;

<SRC-ADDR> – IP-адрес отправителя, в качестве данного адреса может использоваться любой IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-IPV6-ADDR> – IPv6-адрес отправителя, в качестве данного адреса может использоваться любой IPv6-адрес маршрутизатора, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – имя интерфейса маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля маршрутизатора, через который будут отправлены пакеты, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<DSCP> – DSCP-приоритет в соответствии с RFC 2474, принимает значение [0..63], значение по умолчанию 0;

<PORT> – номер TCP/UDP-порта, принимает значение [1..65535], значение по умолчанию 53 для UDP и 80 для TCP;

<GW-ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности;

<GWIPV6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. При указании данного параметра в исходящий пакет добавляется IP source routing опция, которая сообщает маршрутизатору, через какой шлюз должен маршрутизироваться пакет в сети. На большинство маршрутизаторов отключена маршрутизация по данной опции из соображений безопасности.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# traceroute 192.168.27.128
traceroute to 192.168.27.128 (192.168.27.128), 30 hops max, 60 byte packets
1 192.168.16.1 (192.168.16.1)  1.240 ms  1.546 ms  1.883 ms
2 192.168.27.128 (192.168.27.128)  0.451 ms  0.437 ms  0.411 ms
```

uptime

Данной командой осуществляется просмотр продолжительности времени работы устройства.

Синтаксис

```
uptime
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# uptime
System uptime (d,h:m:s):         00,00:26:35
```

verify

Данной командой выполняется расчет хеш-суммы для отдельного файла с использованием указанного алгоритма хеширования.

Синтаксис

```
verify <ALGORITHM> <FILE>
```

Параметры

<ALGORITHM> – алгоритм хеширования, принимает значения [md5, sha2-256, sha2-512].

<FILE> – путь и имя файла для проверки, может принимать следующие значения:

- usb://usb_name:/PATH
- mmc://mmc_name:/PATH
- system:candidate-config

- system:running-config
- system:factory-config
- system:default-config
- system:firmware-image-1
- system:firmware-image-2
- system:boot-1
- system:boot-2
- flash:critlog/FILE
- flash:syslog/FILE
- flash:backup/FILE
- flash:data/FILE
- tmpsys:syslog/FILE

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# verify md5 system:firmware-image-1
system:firmware-image-1 16ef38a292e96ce972e910da6db2d1f4
```

verify filesystem

Данной командой запускается процесс расчета хеш-сумм для всех системных файлов маршрутизатора и сравнения с эталонными значениями. В результате выводится информация о соответствии рассчитанных хеш-сумм эталонным.

Синтаксис

```
verify filesystem [detailed]
```

Параметры

detailed – ключ, отвечающий за подробный вывод информации о проверке каждого файла.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# verify filesystem
Total:          2949
Success:        2949
Changed:        0
Deleted:        0


Filesystem verification success.
```

verify storage-device

Данной командой запускается процесс проверки внешнего носителя на наличие ошибок.

Синтаксис

```
verify storage-device { usb-dev://<ID> | usb://<USB-NAME> | mmc }
```

Параметры

<ID> – номер подключенного USB-носителя по порядку;

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды show storage-devices.

mmc – ключ для проверки установленного mmc-носителя.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# verify storage-device usb://BC1E-2E16
Device verify
Do you really want to continue? (y/N): y
CP437: Invalid argument
fsck.fat 4.0 (2016-05-06)
/dev/sda2: 14 files, 63908/255496 clusters
Device verification success.
```

5. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ И КОНФИГУРАЦИЕЙ

archive

Данной командой осуществляется переход в режим настройки параметров резервирования конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для параметров резервирования конфигурации.

Синтаксис

```
[no] archive
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# archive
rtt(config-archive)#
```

auto

Данной командой включается режим создания файла резервной конфигурации на удаленном сервер и/или локально через указанный промежуток времени (раздел time-period).

Использование отрицательной формы команды (no) выключает режим отправки через указанный промежуток времени.

Синтаксис

```
[no] auto
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# auto
```

banner exec

Данной командой задаётся текстовое сообщение, выводимое после успешной аутентификации пользователя на маршрутизаторе.

Использование отрицательной формы команды (no) удаляется текстовое сообщение, выводимое после успешной аутентификации пользователя на маршрутизаторе.

Синтаксис

```
banner exec <BANNER>
no banner exec
```

Параметры

<BANNER> – текстовое сообщение, задаётся в кавычках строкой до 2047 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# banner exec '              ----- Hello! This is an Rusteletech router.-----
\n----- Unauthorized configuration changes are prosecuted by the law of the Russian
Federation. -----'
```

banner login

Данной командой задаётся текстовое сообщение, выводимое до аутентификации пользователя на маршрутизаторе.

Синтаксис

```
banner login <BANNER>
no banner login
```

Параметры

<BANNER> – текстовое сообщение, задаётся в кавычках строкой до 2047 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# banner login '            ----- Hello! This is an Rusteletech router.-----
\n----- Unauthorized connection is prosecuted by the law of the Russian Federation. -----'
```

Управление программным обеспечением и конфигурацией

Команды *boot host auto-config* и *boot host auto-update* отвечают за управление функции автоматического получения конфигурации и прошивки в рамках работы DHCP. Включение данной функции имеет смысл при выполнении следующих условий:
1. На устройстве сконфигурирован хотя бы один включенный интерфейс с активным DHCP-клиентом.
2. На DHCP клиенте включена опция *tftp-server-name*.

boot host auto-config

Данная команда включает загрузку конфигурации маршрутизатора при помощи протокола DHCP Option 54 (IP-адрес TFTP-сервера) и 67 (имя файла конфигурации).

Использование отрицательной формы команды (no) отключает загрузку конфигурации маршрутизатора при помощи протокола DHCP Option 66 и 67.

Синтаксис

```
[no] boot host auto-config
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Значение по умолчанию

Отключено.

Командный режим

CONFIG

Пример

```
rtt(config)# boot host auto-config
```

boot host auto-update

Данная команда включает загрузку ПО маршрутизатора при помощи протокола DHCP Option 54 (IP-адрес TFTP-сервера) и имя файла ПО в поле FILE в DHCP OFFER. При использовании совместно с boot host auto-config обновление ПО имеет более высокий приоритет.

Использование отрицательной формы команды (no) отключает загрузку ПО маршрутизатора при помощи протокола DHCP Option 150 и 67.

Синтаксис

```
[no] boot host auto-update
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Значение по умолчанию

Отключено.

Командный режим

CONFIG

Пример

```
rtt(config)# boot host auto-update
```

boot system

Данная команда служит для выбора активного образа программного обеспечения, загруженного на устройство.

Синтаксис

```
boot system <IMAGE>
```

Параметры

<IMAGE> – название образа программного обеспечения, который будет загружаться на устройство:

- image-1 – следующая загрузка устройства будет выполнена из первого образа ПО;
- image-2 – следующая загрузка устройства будет выполнена из второго образа ПО;
- inactive - следующая загрузка устройства будет выполнена с неактивного образа ПО.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# boot system image-2
```

by-commit

Данной командой включается режим отправки файла конфигурации на сервер резервирования после удачного применения конфигурации.

Использование отрицательной формы команды (no) выключает режим отправки после удачного применения конфигурации.

Синтаксис

```
[no] by-commit
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# by-commit
```

clear storage-device

Данная команда служит для форматирования всего устройства хранения данных или конкретной партиции. Если жесткий диск без разделов, то для форматирования используется команда *clear storage-device hdd-dev://*.

Синтаксис

```
clear storage-device { usb-dev://<SLOT_NUMBER> <NEW_PARTITION> |
usb://<PARTITION>:/ <NEW_PARTITION> | mmc <NEW_PARTITION> | hdd-dev://<SLOT_NUMBER>
<NEW_PARTITION> | hdd://<PARTITION>:/ <NEW_PARTITION> }
```

Параметры

<SLOT_NUMBER> – номер слота, для WLC всегда 1;

<NEW_PARTITION> – наименование новой партиции, задается строкой от 1 до 15 символов;

<PARTITION> – наименование партиции, которую требуется отформатировать, задается строкой от 1 до 15 символов.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
wlc# clear storage-device hdd-dev://1 new_hdd
Formatting will erase all data on this device
Do you really want to continue? (y/N): y
mke2fs 1.43.1 (08-Jun-2016)
Device clear success.
wlc# 2024-09-16T07:04:18+00:00 %HDD-I-CHANGE: 'new_hdd' has been inserted!
```

commit

Данная команда позволяет применить (сделать действующими) изменения конфигурации. RUNNING-конфигурация замещается конфигурацией CANDIDATE. Для того чтобы примененные изменения стали постоянно действующими, эту операцию необходимо подтвердить командой *confirm* в течение времени, не превышающего время действия таймера подтверждения (по умолчанию 600 секунд, изменяется при помощи команды system config-confirm timeout).

Синтаксис

```
commit [ comment <COMMENT> ] [ confirm-timeout <TIME> ]
```

Параметры

<COMMENT> – комментарий к данному изменению рабочей конфигурации. Задаётся текстовой строкой до 31 символа;

<TIMEOUT> – интервал времени в секундах после ввода команды *commit* до восстановления предыдущей версии конфигурации при отсутствии команды *confirm*, принимает значение в секундах [120..86400].

Значение по умолчанию

<COMMENT> – пустой;

<TIMEOUT> – наследуется значение, установленное командой system config-confirm timeout.

Необходимый уровень привилегий

10

Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

Пример

```
rtt# commit
```

commit check

Данная команда позволяет провалидировать конфигурацию CANDIDATE, пока изменения не применены.

Синтаксис

```
commit check
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# commit check
```

confirm

Команда предназначена для подтверждения применения конфигурации. Если в течение заданного времени (по умолчанию 600 секунд, изменяется при помощи команды system config-confirm timeout) после применения конфигурации командой *commit* не было введено подтверждение, произойдет автоматический откат на действующую ранее конфигурацию. Автоматическая система откатов предотвращает ситуации потери связи с устройством.

Синтаксис

```
confirm
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Значение по умолчанию

110

Командный режим

ROOT

CHANGE-EXPIRED-PASSWORD

Пример

```
rtt# confirm
```

copy

Данная команда служит для копирования файлов между различными источниками и получателями.

Синтаксис

```
copy <SOURCE> <DESTINATION>
```

Параметры

<SOURCE> – источник, задаётся в виде:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой ip ftp client username, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>

  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip ssh client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в ip ssh client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip ssh client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в ip ssh client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http:// <ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ipv6> – IPv6-адрес HTTPS-сервера;

  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- usb://usb_name:/<path>
  - usb_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <path> – путь к файлу на USB-носителе.
- mmc://mmc_name:/<path>
  - mmc_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <path> – путь к файлу на MMC-носителе.
- system:factory-config – заводская конфигурация;
- system:default-config – конфигурация по умолчанию (пустая);
- system:running-config – текущая конфигурация;
- system:candidate-config – конфигурация, которая будет применена после выполнения команды *commit*;
- system:firmware – программное обеспечение устройства. Копирование производится с неактивного образа программного обеспечения устройства;
- system:boot-1 – первичный загрузчик устройства (SBI, bl1, x-loader);
- system:boot-2 – вторичный загрузчик устройства (U-Boot, boot)ж
- system:licence – функциональная лицензия устройства;
- flash:critlog/FILE – папка для сохранения сообщений ядра за все время работы устройства;
- flash:syslog/FILE – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- tmpsys:syslog/FILE – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- flash:backup/FILE – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:ta/FILE – папка для хранения TA-ключей OpenVPN.

<DESTINATION> – назначение, задаётся в виде:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;

  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой ip sftp client password, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в **ip scp client username**);
  - <password> – пароль (настроить пароль по умолчанию можно командой ip ssh client password, описанной в **ip scp client password**);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.

- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip scp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в ip scp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http:// <ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ipv6> – IPv6-адрес HTTPS-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- usb://usb_name:/PATH
  - usb_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <path> – путь к файлу на USB-носителе.

- mmc://mmc_name:/PATH:
  - mmc_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <path> – путь к файлу на MMC-носителе.
- system:candidate-config – конфигурация, которая будет применена после выполнения команды *commit*;
- system:firmware – программное обеспечение устройства. Копирование производится с неактивного образа программного обеспечения устройства;
- system:boot-2 – вторичный загрузчик устройства (U-Boot, boot);
- system:licence – функциональная лицензия устройства;
- system:boot-licence – загрузочная лицензия устройства;
- system:cluster-unit-licences – папка для хранения лицензий устройств в кластере;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:ta/FILE – папка для хранения TA-ключей OpenVPN.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример 1

```
rtt# copy tftp://10.100.100.1:/rtt.cfg system:candidate-config
```

Пример 2

```
rtt# copy tftp://10.100.100.1:/crl.pem crypto:crl/crl.pem
```

count-backup

Данной командой устанавливается максимальное количество локально сохраняемых резервных копий конфигураций.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
count-backup <NUM>
no count-backup
```

Параметры

<NUM> – максимальное количество локально сохраняемых резервных копий конфигураций. Принимает значения в диапазоне [1..100].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# count-backup 20
```

delete

Данная команда служит для удаления лицензий, сертификатов, ключей.

Синтаксис

```
delete <FILE>
```

Параметры

<FILE> – тип файла, может принимать следующие значения (при удалении из папки необходимо указать название файла):

- flash:backup/FILE – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:critlog/FILE – папка для сохранения сообщений ядра за все время работы устройства;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;
- flash:syslog/FILE – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- tmpsys:syslog/FILE – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;
- crypto:ta/FILE – папка для хранения TA-ключей OpenVPN;
- system:licence – лицензия для активации функционала, требующего лицензию;

- system:boot- licence – лицензия устройства, применяющаяся при запуске вторичного загрузчика;
- system:cluster-unit-licences/[Serial number] – папка для хранения лицензий устройств в кластере;
- usb://usb_name:/FILE
    - usb_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
    - <FILE> – путь к файлу и его имя на USB-носителе.
- mmc://mmc_name:/<FILE>
    - mmc_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
    - <FILE> – путь к файлу и его имя на MMC-носителе.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# delete crypto:dh/dh.key
```

dir

Данная команда позволяет отобразить содержимое внешних носителей информации (USB/MMC-карт памяти) и локальных разделов маршрутизатора.

Синтаксис

```
dir <PATH>
{ usb://<USB-device-name>/<PATH> | mmc://<MMC-device-name>/<PATH> |
flash:backup/FILE | flash:data/FILE }
```

Параметры

<PATH> – имя локального раздела или путь до папки на внешнем носителе. Принимает значения:

- flash:critlog/ – папка для сохранения сообщений ядра за все время работы устройства;
- flash:syslog/ – папка для сохранения логов текущей сессии, сохраняется после перезагрузки;
- flash:backup/ – папка для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:data/ – папка для скачивания файлов с маршрутизатора;
- tmpsys:syslog/ – папка для сохранения логов текущей сессии, не сохраняется после перезагрузки;
- crypto:cert/FILE – папка для хранения сертификатов;
- crypto:crl/FILE – папка для хранения списков отозванных сертификатов удостоверяющих центров;
- crypto:private-key/FILE – папка для хранения приватных ключей;
- crypto:public-key/FILE – папка для хранения публичных ключей;
- crypto:dh/FILE – папка для хранения ключей Диффи-Хеллмана;
- crypto:pfx/FILE – папка для хранения контейнеров PKCS#12;
- crypto:csr/FILE – папка для хранения запросов на сертификацию X.509;

- crypto:ta/FILE – папка для хранения TA-ключей OpenVPN;
- usb://usb_name:/PATH
  - usb_name – имя, назначенное USB-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <PATH> – путь к файлу на USB-носителе.
- mmc://mmc_name:/PATH
  - mmc_name – имя, назначенное MMC-носителю. Возможно посмотреть командой *show storage-devices* (см. раздел show storage-devices);
  - <PATH> – путь к файлу на MMC-носителе.
- flash:backup/FILE – файл для сохранения резервных копий текущих конфигураций маршрутизатора;
- flash:data/FILE – папка для скачивания файлов с маршрутизатора;

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# dir mmc://EF28-D074
Name                                    Type         Size
--------------------------------------  ----------   --------   --
rtt-1.4.0-build21.uboot                 File         0.00       B
.Trash-1000                             Directory    0.00       B
```

ip ftp client password

Данной командой определяется пароль по умолчанию для операций копирования по протоколу FTP.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
ip ftp client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
[no] ftp client password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [2..32] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip ftp client password test
```

ip ftp client username

Данной командой определяется имя пользователя по умолчанию для операций копирования по протоколу FTP.

Использование отрицательной формы команды (no) удаляет имя пользователя.

Синтаксис

```
ip ftp client username <NAME>
no ftp client username
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip ftp client username test
```

ip ftp source-ip

Данной командой указывается адрес, с которого будет устанавливаться соединение с FTP-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip ftp source-ip <ADDR>
[no] ip ftp source-ip
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]

Значение по умолчанию

IP-адрес интерфейса, с которого отправляются пакеты на FTP-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip ftp source-ip 192.168.1.57
```

ip scp client password

Данная команда задаёт пароль пользователя для клиента протокола SCP.

Использование отрицательной формы команды (no) удаляет пароль пользователя для клиента протокола SCP.

Синтаксис

```
ip scp client password { <TEXT> | encrypted < ENCRYPTED-TEXT > }
no ip scp client password
```

Параметры

<TEXT> – строка [1..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32].

Значение по умолчанию

Имя пользователя не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip scp client password 123456789
```

ip scp client source-ip

Данной командой указывается адрес, с которого будет устанавливаться соединение с SCP-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip scp client source-ip [ vrf <VRF> ] <ADDR>
no ip scp client source-ip [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip scp client source-ip 192.168.1.58
```

ip scp client username

Данная команда задаёт имя пользователя для клиента протокола SCP.

Использование отрицательной формы команды (no) удаляет имя пользователя для клиента протокола SCP.

Синтаксис

```
ip scp client username <USERNAME>
no ip scp client username
```

Параметры

<USERNAME> – имя пользователя, задаётся строкой до 31 символа.

Значение по умолчанию

Имя пользователя не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip scp client username rttuser
```

ip sftp client password

Данная команда задаёт пароль пользователя для клиента протокола SFTP.

Использование отрицательной формы команды (no) удаляет пароль пользователя для клиента протокола SFTP.

Синтаксис

```
ip sftp client password { <TEXT> | encrypted < ENCRYPTED-TEXT > }
no ip sftp client password
```

Параметры

<TEXT> – строка [1..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32].

Значение по умолчанию

Имя пользователя не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip sftp client password 123456789
```

ip sftp client source-ip

Данной командой указывается адрес, с которого будет устанавливаться соединение с SFTP-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip sftp client source-ip [ vrf <VRF> ] <ADDR>
no ip sftp client source-ip [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip sftp client source-ip 192.168.1.57
```

ip sftp client username

Данная команда задаёт имя пользователя для клиента протокола SFTP.

Использование отрицательной формы команды (no) удаляет значения имени пользователя для клиента протокола SFTP.

Синтаксис

```
ip sftp client username <USERNAME>
no ip sftp client username
```

Параметры

<USERNAME> – имя пользователя, задаётся строкой до 31 символа.

Значение по умолчанию

Имя пользователя не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip sftp client username rttuser
```

ip tftp client timeout

Данной командой настраивается время ожидания ответа от TFTP-сервера для TFTP-клиента на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip tftp client timeout <TIMER>
no ip tftp client timeout
```

Параметры

<TIMER> – максимальное значение времени ожидания ответа от TFTP-сервера в секундах, принимает значение [10..60].

Значение по умолчанию

10

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip tftp client timeout 15
```

merge

Данная команда служит для объединения локального или загружаемого файла конфигурации с candidate-config.

Синтаксис

```
merge <SOURCE> system:candidate-config
```

Параметры

<SOURCE> – источник, задаётся в виде:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в **ip ftp client password**);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);

  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip scp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в ip scp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.
- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip scp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password,* описанной в ip scp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.
- http://<ip>[<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- http://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTP-сервере, где:
  - <ipv6> – IPv6-адрес HTTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ip>[<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;
  - <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
  - <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
  - <ip> – IP-адрес HTTP-сервера;
  - <port> – порт, на котором запущен HTTPS-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на HTTP-сервере.
- https://[ca/<CA-FILE>][:server-crt/<SC-FILE>][:server-key/<SK-FILE>]@<ipv6>[%<interface>][<port>]:/<path> – адрес файла на HTTPS-сервере, где:
  - <CA-FILE> – имя файла сертификата удостоверяющего сервера в соответствующем разделе памяти маршрутизатора;

- <SC-FILE> – имя файла публичного сертификата сервера в соответствующем разделе памяти маршрутизатора;
- <SK-FILE> – имя файла приватного ключа сервера в соответствующем разделе памяти маршрутизатора;
- <ipv6> – IPv6-адрес HTTPS-сервера;
- <interface> – исходящий сетевой интерфейс для link-local адресов;
- <port> – порт, на котором запущен HTTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
- <path> – относительный путь к файлу на TFTP-сервере.

- flash:data/FILE – папка для скачивания файлов с маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# merge tftp://10.100.100.1/rtt.cfg system:candidate-config
```

path

Данной командой определяются протокол, адрес сервера, а также расположение и префикс имени файла на сервере. При выполнении резервирования к префиксу имени файла добавляется текущее время и дата в формате ГГГГММДД_ЧЧММСС.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

```
path <PATH>
no path
```

Параметры

<PATH> – формат пути до папки на удаленном сервере по протоколам tftp/ftp/sftp/scp в одном из следующих форматов:

- tftp://<ip>[<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ip> – IP-адрес TFTP-сервера;
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на TFTP-сервере.
- tftp://<ipv6>[%<interface>][<port>]:/<path> – адрес файла на TFTP-сервере, где:
  - <ipv6> – IPv6-адрес TFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;

  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на TFTP-сервере.
- ftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес FTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- ftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес FTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ftp client username*, описанной в ip ftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ftp client password*, описанной в ip ftp client username);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на FTP-сервере.
- sftp://[<user>[:<password>]@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес SFTP-сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный путь к файлу на SFTP-сервере.
- sftp://[<user>[:<password>]@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес SFTP-сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip sftp client username*, описанной в ip sftp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip sftp client password*, описанной в ip sftp client password);
  - <port> – порт, который слушает SFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный путь к файлу на SFTP-сервере.
- scp://[<user>:<password>@]<ip>[<port>]:/<path>
  - <ip> – IP-адрес сервера;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip scp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password*, описанной в ip scp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IP-адреса символом «#» или «:»;
  - <path> – относительный или абсолютный путь к файлу на сервере.

- scp://[<user>:<password>@]<ipv6>[%<interface>][<port>]:/<path>
  - <ipv6> – IPv6-адрес сервера;
  - <interface> – исходящий сетевой интерфейс для link-local адресов;
  - <user> – имя пользователя (настроить имя пользователя по умолчанию можно командой *ip ssh client username*, описанной в ip scp client username);
  - <password> – пароль (настроить пароль по умолчанию можно командой *ip ssh client password,* описанной в ip scp client password);
  - <port> – порт, который слушает TFTP-сервер, отделяется от IPv6-адреса символом '#' или ':' (в данном случае IPv6-адрес должен быть заключен в квадратные скобки «[]»);
  - <path> – относительный или абсолютный путь к файлу на сервере.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# path tftp://10.10.10.1:/rtt/config
```

restore

Данная команда позволяет отменить примененную, но неподтвержденную конфигурацию и вернуться к последней подтвержденной. Команда применяется ко всей конфигурации устройства. Отмена изменений может быть выполнена только до ввода команды *confirm*. При выполнении команды *restore* происходит потеря неподтвержденной конфигурации.

Синтаксис

```
restore
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# restore
```

Осуществлен возврат к последней подтвержденной конфигурации.

rollback

Данная команда позволяет изменить содержимое candidate-config.

Команда применяется ко всей конфигурации устройства.

Синтаксис

```
rollback [to-backup <FILENAME>]
```

Параметры

<FILENAME> – имя файла ранее сохраненной конфигурации из раздела flash:backup/ маршрутизатора.

Значение по умолчанию

Без использования ключа "to-backup" в candidate-config копируется текущая "running-config".

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# rollback
```

Произведена отмена всех непримененных изменений в конфигурации.

save

Команда служит для сохранения CANDIDATE-конфигурации в постоянную память устройства.

Синтаксис

```
save
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# save
```

Сохранение текущей конфигурации на Flash-память устройства.

show bootvar

Данная команда служит для просмотра информации об образах программного обеспечения, загруженных на устройство.

Синтаксис

```
show bootvar
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show bootvar
Image   Version                      Date                  Status         After reboot
-----   --------------------------   -------------------   ------------   --------
1       1.24.0 build 1[5cd22b8]      2024-08-01 18:00:47    Not Active
2       1.24.0 build 1[5cd22b8]      2024-08-01 18:00:47    Active            *
```

show candidate-config

Данной командой осуществляется просмотр конфигурации устройства, которая будет установлена после применения настроек (команда *commit*).

Синтаксис

```
show candidate-config [ full ] [ <SECTION> ]
```

Параметры

full – ключ для отображения в конфигурации значений всех параметров, в том числе и ненастроенных.

<SECTION> – раздел конфигурации:

- aaa – настройка параметров аутентификации, авторизации и учета;
- access-list – конфигурация списков доступа;
- bridges – конфигурация сетевых мостов;
  - [<NUM>] – номер сетевого моста.
- channel-group – конфигурация группы агрегации каналов;
- clock – конфигурация системных часов маршрутизатора и NTP-протокола;
- dhcp – конфигурация DHCP-сервера, клиента и Relay-агента;
- domain – конфигурация domain lookup;
- content-provider – конфигурация источника обновлений правил, распространяемых по коммерческой лицензии;
- crypto-pki [<SUBSECTION>] – конфигурация инфраструктуры открытых ключей;
  - server – конфигурация PKI-сервера;
  - trustpoint [NAME] – конфигурация PKI-клиентов;
    - [NAME] – имя PKI-клиента.
- dual-homing – конфигурация сервиса Dual Homing[1];
- extended – расширенный вывод конфигурации;
- hostname – сетевое имя маршрутизатора;
- interfaces [<IF>] – конфигурация интерфейсов:
  - <IF> – наименование интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.
- ip-address – конфигурация IP-интерфейсов;
- ipv6 [<SUBSECTION>] – конфигурация IPv6:
  - address – конфигурация IPv6-интерфейсов;
  - dhcp [<SUBSECTION>] – конфигурация IPv6 DHCP-сервисов:
  - client – конфигурация IPv6 DHCP-сервера;
    - relay – конфигурация IPv6 DHCP Relay-агента;
    - server – конфигурация IPv6 DHCP-клиента.
  - routing [<SUBSECTION>] – конфигурация IPv6-маршрутизации:
    - bfd – конфигурация протокола IPv6 BFD;
    - bgp – конфигурация протокола IPv6 BGP;
    - ospf – конфигурация протокола OSPFv3;
    - prefix-lists – конфигурация IPv6 префикс-листов;
    - rip – конфигурация протокола RIP;
    - static – конфигурация статических маршрутов.
  - vrrp – конфигурация IPv6 VRRP-протокола.
- dialplan – конфигурация плана нумерации;
- lldp – конфигурация протокола LLDP;
- mac-address-table – конфигурация таблицы MAC-адресов;
- mailservers – конфигурация почтовых серверов и доменов;
- mdns – функционала mDNS;
- mirroring – конфигурация зеркалирования;
- mpls – конфигурация протоколов технологии MPLS;
- mdns – конфигурация функционала mDNS;
- multiwan – конфигурация сервиса резервирования и балансировки WAN-интерфейсов;
- nat [<SUBSECTION>] – конфигурация сервиса NAT:
  - source – конфигурация сервиса Source NAT;
  - destination – конфигурация сервиса Destination NAT;

    - alg – конфигурация алгоритма NAT.
- netflow – конфигурация Netflow-протокола;
- object-groups [<TYPE> [<NAME>]] – конфигурация профилей;
    - <TYPE> – тип профиля, принимает значения:
        - content-filter;
        - email;
        - network;
        - address-port;
        - application;
        - mac;
        - service;
        - url;
        - domain-name.
    - <NAME> – имя профиля заданного типа.
- port-security – конфигурация Port Security (только для R800);
- qos – конфигурация QoS;
- remote-access [<SUBSECTION>] – конфигурация профилей L2TP over IPsec и PPTP-серверов;
    - l2tp – конфигурация профилей L2TP over IPsec-серверов;
    - pptp – конфигурация профилей PPTP-серверов;
    - openvpn – конфигурация профилей OpenVPN-серверов.
- remote-client – конфигурация удаленного доступа (SSH, Telnet, etc.);
- rmon – конфигурация RMON;
- routing [<SUBSECTION>] – конфигурация маршрутизации:
    - bfd – конфигурация протокола BFD;
    - bgp – конфигурация протокола BGP;
    - isis – конфигурация протокола IS-IS;
    - key-chains – конфигурация ключей аутентификации;
    - ospf – конфигурация протокола OSPF;
    - prefix-list – конфигурация префикс-листов;
    - rip – конфигурация протокола RIP;
    - route-maps – конфигурация маршрутных карт
        - [<RM-NAME>] – имя маршрутной карты.
    - static – конфигурация статических маршрутов.
- security [<SUBSECTION>] – конфигурация сервисов IPsec VPN и Firewall;
    - ike – конфигурация IKE;
    - ipsec – конфигурация IPsec;
    - zone – конфигурация зон Firewall;
    - zone-pair – конфигурация переходов между зонами Firewall.
- sip – конфигурация протокола SIP;
    - profile – конфигурация SIP-профилей;
    - service – конфигурация SIP.
- sflow – конфигурация sFlow-протокола;
- snmp – конфигурация SNMP-сервера;
- spanning-tree – конфигурация протоколов семейства Spanning Tree;
- sla – конфигурация сервиса IP SLA;
- system – конфигурация общесистемных параметров;
- syslog – конфигурация сервиса Syslog;
- tracks – конфигурация Tracking-объектов.
- tunnels [<TYPE> [<NUM>]] – конфигурация туннелей:
    - <TYPE> – тип туннеля, принимает значения:

        - gre – конфигурация GRE-туннелей;
        - ip4ip4 – конфигурация IPv4 over IPv4-туннелей;
        - l2tp – конфигурация L2TP-туннелей;
        - l2tpv3 – конфигурация L2TPv3-туннелей;
        - lt – конфигурация логических туннелей;
        - softgre – конфигурация SoftGRE-туннелей (только для R800);
        - vti – конфигурация VTI-туннелей;
        - pptp – конфигурация PPTP-клиентов;
        - pppoe – конфигурация PPPoE-клиентов;
        - l2tp – конфигурация L2TP-клиентов;
        - openvpn – конфигурация OpenVPN-клиентов.
    - <NUM> – номер туннеля данного типа в конфигурации маршрутизатора.
- vlans – конфигурация VLAN;
- voice – конфигурация голосовых сервисов;
- vrf – конфигурация VRF;
- vrrp – конфигурация VRRP-протокола;
- wireless-controller – конфигурация параметров Wi-Fi контроллера;
- zabbix – конфигурация Zabbix-агента.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show candidate-config
ntp enable
ntp broadcast-client enable
syslog max-files 3
syslog file-size 512
syslog file default info
vlan 2
exit
security zone trusted
exit
security zone untrusted
exit
object-group service telnet
  port-range 23
exit
object-group service ssh
  port-range 22
exit
object-group service dhcp_server
  port-range 67
exit More? Enter - next line; Space - next page; Q - quit; R - show the rest.
```

show configuration changes

Данной командой осуществляется просмотр различий между конфигурационными файлами.

Синтаксис

```
show configuration changes [<CONFIG> < CONFIG>]
```

Параметры

<CONFIG> – конфигурационный файл для сравнения. Могут принимать значения:

- candidate-config;
- running-config;
- factory-config;
- default-config;
- flash:backup/FILE.

Значение по умолчанию

Без указания <CONFIG> отображается разница между running-config и candidate-config.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt(config-)# show configuration changes
+ interface gigabitethernet 1/0/1.100
+   ip firewall disable
+   ip address 10.54.22.1/24
+ exit
```

show crypto certificates

Данная команда выводит информацию о количестве сертификатов.

Синтаксис

```
show crypto certificates [ <CERTIFICATE-TYPE> [ <FILE> ] ]
```

Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- cert – X.509 сертификаты;
- dh – ключ Диффи-Хеллмана;

- private-key – приватные ключи;
- public-key – публичные ключи;
- ta – HMAC-ключ;
- crl – отозванные сертификаты;
- csr – запросы на сертификацию X.509;
- pfx – PKSC12-контейнер.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show crypto certificates
Type           Total
------------   -------
cert           16
crl            1
csr            5
dh             3
pfx            31
private-key    13
public-key     3
ta             2
```

show running-config

Данная команда служит для просмотра текущей конфигурации устройства.

Синтаксис

```
show running-config [<SECTION>] [ full ]
```

Параметры

<SECTION> – раздел конфигурации, описание приведено в разделе show candidate-config.

full – ключ для отображения в конфигурации значений всех параметров, в том числе и ненастроенных.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show running-config syslog
syslog max-files 3
syslog file-size 512
syslog file default info
syslog console info
```

show storage-devices

Данной командой выводится информация о подключенных внешних носителях информации (USB/MMC-карт памяти).

Синтаксис

```
show storage-devices { usb | mmc | hdd}
```

Параметры

usb – USB-накопитель информации;

mmc – SD/MMC-карта памяти.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show storage-devices mmc
Name                            Total, MB    Used, MB     Free, MB
------------------------------  ----------   ----------   ----------
EF28-D074                       99.79        72.64        27.15
```

show usb

Данной командой выводится информация о подключенных USB-носителях.

Синтаксис

```
show usb
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show usb
USB device   Vendor      Product ID          Product version
----------   ---------   -----------------   ---------------
1-1          Flash       USB Disk            6.80
```

show version

Данная команда служит для просмотра текущей версии программного обеспечения и аппаратной части устройства.

Синтаксис

```
show version [ bl | hw | sw | voip ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<bl> – этот параметр позволяет просматривать только версию boot. Недоступно на vRTT;

<hw> – этот параметр позволяет просматривать только версию аппаратной части устройства. Недоступно на vRTT;

<sw> – этот параметр позволяет просматривать только версию программного обеспечения;

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения версии устройства, на котором вводится команда, в формате вывода без включенного кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show version
Boot version:
  1.24.0.1 (2024-08-01 13:40:59)
SW version:
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
```

```
HW version:
  1v7

rtt# show version bl
Boot version:
  1.24.0.1 (2024-08-01 13:40:59)

rtt# show version hw
HW version:
  1v7

rtt# show version sw
SW version:
  1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
```

time-period

Данной командой задаётся период времени, по истечении которого будет осуществляться автоматическое резервирование конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
time-period <TIME>
no time-period
```

Параметры

<TIME> – периодичность автоматического резервирования конфигурации, принимает значение в минутах [1..525600].

Значение по умолчанию

720 минут.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# time-period 1440
```

type

Данной командой устанавливается тип сохранения резервных конфигураций маршрутизатора.

При использовании отрицательной формы команды (no) устанавливается режим по умолчанию.

Синтаксис

```
type <TYPE>
no type
```

Параметры

<TYPE> – тип сохранения резервных конфигураций маршрутизатора. Принимает значения:

- local – сохранения резервных конфигураций происходит в раздел flash:backup/ с именем файла вида "config_YYYYMMDD_HHMMSS_username;" где:
  - YYYY – год, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - MM – месяц, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - DD – день, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - HH – час, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - MM – минута, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации;
  - SS – секунда, согласно системным часам маршрутизатора на момент записи резервной копии конфигурации.
  - username – имя учетной записи, в рамках сессии которой был выполнен «commit».
- remote – сохранения резервных конфигураций происходят на удаленный сервер;
- both – сохранения резервных конфигураций в раздел flash:backup и на удаленный сервер.

Значение по умолчанию

remote

Необходимый уровень привилегий

15

Командный режим

CONFIG-ARCHIVE

Пример

```
rtt(config-archive)# type both
```

unmount storage-device

Данной командой производится отключение внешнего накопителя.

Синтаксис

```
unmount storage-device { usb | mmc | hdd }
```

Параметры

usb – USB-накопитель информации;

mmc – SD-/MMC-карта памяти.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show storage-devices usb
rtt# 2024-04-27T08:55:40+00:00 %USBFLASH-I-CHANGE: 'D2E5-006B' has been inserted!
```

update crypto default

Данной командой обновляются сертификаты.

Синтаксис

```
update crypto default <CERTIFICATE-TYPE> [invalid-after] [valid-after]
```

Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- dh – ключ Диффи-Хеллмана;
- ca – корневой сертификат;
- cert – X.509-сертификат.

invalid-after – установка даты, до конца которой будет действовать сертификат;

valid-after – установка даты, от начала которой будет действовать сертификат.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# update crypto default ca invalid-after 00:00:00 01 january 2025
rtt# show crypto certificates cert default_ca.pem | i after
    Valid after:                                2024-06-13 12:27:57
    Invalid after:                              2025-01-01 00:00:00
```

6. НАСТРОЙКА TFTP-СЕРВЕРА

dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов TFTP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
dscp <DSCP>
no dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значение в диапазоне [0..63].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-TFTP-SERVER

Пример

```
rtt(config-tftp-server)# dscp 22
```

enable

Данной командой включается TFTP-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает TFTP-сервер.

Синтаксис

```
[no] enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TFTP-SERVER

Пример

```
rtt(config-tftp-server)# enable
```

ip tftp server

Данная команда переводит маршрутизатор в режим конфигурирования TFTP-сервера.

Рабочим разделом TFTP-сервера является flash:data/.

Использование отрицательной формы команды (no) отключает TFTP-сервер и удаляет примененные настройки TFTP-сервера.

Синтаксис

```
ip tftp server vrf <VRF>
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать TFTP-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip tftp server
```

local-port

Данной командой определяется порт TFTP-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
local-port <PORT>
```

Параметры

<PORT> – номер порта, принимает значение в диапазоне [1..65535].

Значение по умолчанию

69

Необходимый уровень привилегий

10

Командный режим

CONFIG-TFTP-SERVER

Пример

```
rtt(config-tftp-server)# local-port 6212
```

7. НАСТРОЙКА DNS

domain ip host

Данная команда определяет статическую DNS-запись.

Использование отрицательной формы команды (no) удаляет запись.

Синтаксис

```
domain ip host <NAME> <IP>
no domain ip host { <NAME> <IP> | all }
```

Параметры

<NAME> – имя хоста, задаётся строкой до 253 символов.

<IP> – IP-адрес хоста, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Есть возможность использовать ключ "all" для удаления всех ранее настроенных статических DNS-записей.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# domain ip host Rusteletech.loc 172.16.0.3
```

domain lookup enable

Данная команда включает разрешение DNS-имен.

Использование отрицательной формы команды (no) выключает разрешение DNS-имен.

Синтаксис

```
[no] domain lookup enable [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет включено разрешение DNS-имен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# domain lookup enable
```

domain name

Команда позволяет назначить имя домена для маршрутизатора.

Использование отрицательной формы команды (no) удаляет имя домена для маршрутизатора.

Синтаксис

```
domain name <NAME>
no domain name
```

Параметры

<NAME> – имя домена маршрутизатора, задаётся строкой от 1 до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# domain name Rusteletech.ru
```

domain nameserver

Данная команда определяет IP-адрес DNS-сервера, используемого для разрешения DNS-имен.

Использование отрицательной формы команды (no) удаляет адрес DNS-сервера.

Синтаксис

```
[no] domain nameserver <IP> [vrf <VRF>]
```

Параметры

<IP> – IP-адрес используемого DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет происходить взаимодействие с DNS-сервером.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# domain nameserver 172.16.0.1
```

show dns records

Данной командой выводится кэш dns-записей, которые хранятся на устройстве до истечения TTL.

Синтаксис

```
show dns records [ domain <NAME> | ip <IP> ]
```

Параметры

<NAME> – имя хоста, задаётся строкой до 253 символов;

<IP> – IP-адрес хоста, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show dns records
Domain                      IP addres                 Time to live
                                                        (d,h:m:s)
----------------------   -------------------------   ----------------
Rusteletech.ru                62.109.9.104               00,00:09:51
```

show dns records negative

Данной командой выводится кэш негативных dns-записей, которые хранятся на устройстве до истечения TTL.

Синтаксис

```
show dns records negative [ domain <NAME> ]
```

Параметры

<NAME> – имя хоста, задаётся строкой до 253 символов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show dns records negative
Domain                              IP address          Time to live
                                                         (d,h:m:s)
---------------------------------   -----------------   ----------------
sadafplus.yandex.ru                 213.180.204.242     00,00:09:38
```

show dns servers

Данной командой выводится список настроенных на устройстве dns-серверов.

Синтаксис

```
show dns servers
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show dns servers
DNS server                                  Source
------------------------------------------  ----------------------
198.18.3.59                                 static
```

8. НАСТРОЙКА MDNS-REFLECTOR

clear ip mdns-reflector

Данная команда обновляет данные в кэше mDNS-reflector.

Синтаксис

```
clear ip mdns-reflector
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# clear ip mdns-reflector
```

ip mdns-reflector

Данная команда добавляет интерфейс в конфигурацию сервиса mDNS-reflector. Для запуска сервиса нужно включить в конфигурацию минимум два интерфейса.

Использование отрицательной команды (no) удаляет интерфейс из конфигурации сервиса mDNS-reflector.

Синтаксис

```
ip mdns-reflector
no ip mdns-reflector
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip mdns-reflector
```

ip mdns-reflector services

Данная команда активирует фильтрацию сервисов mDNS. Сервисы, указанные в object-group <NAME>, будут разрешены, все остальные будут запрещены. Если список фильтрации пуст, то будут доступны все сервисы.

Использование отрицательной команды (no) отключает фильтрацию сервисов mDNS.

Синтаксис

```
ip mdns-reflector services <OBJ-GROUP-URL-NAME>
no ip mdns-reflector services
```

Параметры

<OBJ-GROUP-URL-NAME> – имя профиля URL-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt (config)# object-group url test
rtt (config-object-group-url)# url _ssh._tcp.local
rtt (config-object-group-url)# exit
rtt (config)#
rtt (config)# ip mdns-reflector services test
```

show ip mdns-reflector

Данная команда выводит список сервисов из кэша mDNS-reflector.

Синтаксис

```
show ip mdns-reflector [ interfaces <IF> | service <NAME> ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<NAME> – имя профиля URL-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip mdns-reflector
Interface IP address Hostname Service Port
---------- --------------- ------------------- ----------------- ------
br21 172.21.0.10 WEP-12ac.local _brcm-sb._tcp 80
```

```
br21 172.21.0.10 WEP-12ac.local _telnet._tcp 23
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
br21 172.21.0.10 WEP-12ac.local _http._tcp 80
br21 172.21.0.10 WEP-12ac.local _https._tcp 443
gi1/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _eppc._tcp 3031
gi1/0/2.20 172.20.0.11 Book-tester-7.local _rfb._tcp 5900
gi1/0/2.20 172.20.0.11 Book-tester-7.local _smb._tcp 445
gi1/0/2.20 172.20.0.11 Book-tester-7.local _airplay._tcp 7000
gi1/0/2.20 172.20.0.11 Book-tester-7.local _raop._tcp 5000
gi1/0/2.20 172.20.0.11 Book-tester-7.local _teamviewer._tcp 2020
rtt# show ip mdns-reflector service ssh
Interface IP address Hostname Service Port
---------- --------------- ------------------- ----------------- ------
br21 172.21.0.10 WEP-12ac.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _ssh._tcp 22
gi1/0/2.20 172.20.0.11 Book-tester-7.local _sftp-ssh._tcp 22
```

9. НАСТРОЙКА ОБЩЕСИСТЕМНЫХ ПАРАМЕТРОВ

9.1. Общие параметры

alias

Данная команда дает возможность настроить укороченные/специфичные команды в различных командных режимах.

Использование отрицательной формы команды (no) удаляет запись.

Для активации настроенных alias необходимо переподключиться к CLI маршрутизатора.

Синтаксис

```
alias {<ALIAS_NAME>} {<MODE>} {<COMMAND>}
no alias <ALIAS_NAME>
```

Параметры

<ALIAS_NAME> – имя новой команды.

<MODE> – названия командного режима, в котором присутствует новая команда:

- aaa-ldap-profile-configure;
- config;
- config-aaa-das-profile;
- config-aaa-radius-profile;
- config-access-profile;
- config-access-profile-user;
- config-acl;
- config-acl-rule;
- config-address-assignment-pool;
- config-archive;
- config-bgp;
- config-bgp-af;
- config-bgp-aggregate;
- config-bgp-group;
- config-bgp-group-af;
- config-bgp-group-af-vrf;
- config-bgp-listen;
- config-bgp-local-as;
- config-bgp-neighbor;
- config-bgp-neighbor-af;
- config-bgp-neighbor-af-vrf;

- config-bgp-vrf;
- config-bgp-vrf-af;
- config-bgp-vrf-aggregate;
- config-bgp-vrf-group;
- config-bgp-vrf-listen;
- config-bgp-vrf-local-as;
- config-bgp-vrf-neighbor;
- config-bridge;
- config-cellular-modem;
- config-cellular-profile;
- config-class-map;
- config-class-policy-map;
- config-cluster;
- config-cluster-unit;
- config-content-provider;
- config-crypto-sync;
- config-das-server;
- config-dhcp-server;
- config-dhcp-server-failover;
- config-dhcp-server-vendor-specific;
- config-dhcp-vendor-id;
- config-dn;
- config-dnat;
- config-dnat-pool;
- config-dnat-rule;
- config-dnat-ruleset;
- config-failover;
- config-firewall-failover;
- config-gre;
- config-http-profile;
- config-if-e1;
- config-if-gi;
- config-if-loopback;
- config-if-multilink;
- config-if-port-channel;
- config-if-qinq;
- config-if-sub;
- config-ike-gw;
- config-ike-keyring;
- config-ike-policy;
- config-ike-proposal;
- config-ip4ip4;
- config-ips;
- config-ips-auto-upgrade;
- config-ips-category;
- config-ips-category-rule;
- config-ips-category-rule-advanced;
- config-ips-content-filter;
- config-ips-policy;
- config-ips-policy-vendor;

- config-ips-policy-vendor-category;
- config-ips-upgrade-user-server;
- config-ipsec-policy;
- config-ipsec-proposal;
- config-ipsec-vpn;
- config-ipv6-dhcp-server;
- config-ipv6-ospf;
- config-ipv6-ospf-area;
- config-ipv6-ospf-vlink;
- config-ipv6-pl;
- config-ipv6-vrrp;
- config-ipv6-wan-rule;
- config-ipv6-wan-target;
- config-ipv6-wan-target-list;
- config-isis;
- config-key-chain;
- config-key-chain-key;
- config-l2tp;
- config-l2tp-server;
- config-l2tpv3;
- config-l2vpn;
- config-l2vpn-autodiscovery-bgp;
- config-l2vpn-eompls;
- config-l2vpn-pw;
- config-l2vpn-pw-class;
- config-ldap-server;
- config-ldp;
- config-ldp-af;
- config-ldp-af-if;
- config-ldp-neighbor;
- config-licence-manager;
- config-line-console;
- config-line-ssh;
- config-line-telnet;
- config-lt;
- config-mailserver;
- config-mailserver-domain;
- config-mpls;
- config-multipath-ipv6-route;
- config-multipath-route;
- config-net-policy;
- config-netflow-host;
- config-ntp;
- config-object-group-address-port;
- config-object-group-application;
- config-object-group-cf-kaspersky;
- config-object-group-content-filter;
- config-object-group-domain-name;
- config-object-group-email;
- config-object-group-mac;

- config-object-group-network;
- config-object-group-service;
- config-object-group-url;
- config-openvpn;
- config-openvpn-server;
- config-openvpn-server-user;
- config-ospf;
- config-ospf-area;
- config-ospf-vlink;
- config-pki-server;
- config-pl;
- config-policy-map;
- config-ppp-user;
- config-pppoe;
- config-pptp;
- config-pptp-server;
- config-radius-server;
- config-rip;
- config-ripng;
- config-route-map;
- config-route-map-rule;
- config-san;
- config-security-zone;
- config-security-zone-pair;
- config-security-zone-pair-rule;
- config-sflow-host;
- config-sla-responder;
- config-sla-test;
- config-snat;
- config-snat-pool;
- config-snat-rule;
- config-snat-ruleset;
- config-snmp;
- config-snmp-host;
- config-snmp-user;
- config-softgre;
- config-softgre-controller;
- config-subscriber-control;
- config-subscriber-default-service;
- config-subtunnel;
- config-syslog-destination;
- config-syslog-file;
- config-syslog-host;
- config-tacacs-server;
- config-tftp-server;
- config-track;
- config-trustpoint;
- config-user;
- config-vlan;
- config-vrf;

- config-vrrp;
- config-vti;
- config-wan-rule;
- config-wan-target;
- config-wan-target-list;
- config-wireguard;
- config-wireguard-peer;
- config-wireguard-server;
- config-wireguard-server-peer;
- config-zabbix-agent;
- config-zabbix-proxy;
- debug;
- root;
- root-change-expired-password.

<COMMAND> – строка размером 96 символов. Команда CLI. Вводится полностью, без сокращений. Для ввода команды из нескольких слов обрамляется двойными кавычками.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# alias qwe root "show version"
```

hostname

Команда позволяет назначить сетевое имя для маршрутизатора.

Использование отрицательной формы команды (no) устанавливает имя маршрутизатора по умолчанию.

Синтаксис

```
hostname <NAME> [ unit <ID> ]
no hostname [ unit <ID> ]
```

Параметры

<NAME> – сетевое имя маршрутизатора, задаётся строкой до 63 символов;

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Значение по умолчанию

R100/R200/R800/vrtt – в зависимости от модели маршрутизатора.

Командный режим

CONFIG

Пример

```
rtt(config)# hostname router-1.Rusteletech.ru
```

set unit id

Данной командой выставляется номер юнита устройства.

Синтаксис

```
set unit id <ID>
```

Параметры

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# set unit id 2
Unit ID will be 2 after reboot
```

show cpu network-load

Данной командой осуществляется просмотр нагрузки, производимой сетевым трафиком.

Синтаксис

```
show cpu network-load
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cpu network-load
CPU ID    CPU load    Heaviest session          Session
                                                weight
-------   ---------   ----------------------   -------
0         0           --                       0
1         49413       80.88.157.57 ->          9826
                      172.129.22.57
2         46812       80.88.157.75 ->          9895
                      172.129.22.75
3         49229       172.129.22.41 ->         9851
                      80.88.157.41
4         0           --                       0
5         53019       80.88.157.77 ->          9989
                      172.129.22.77
6         39699       80.88.157.79 ->          9863
                      172.129.22.79
7         49726       172.129.22.45 ->         9804
                      80.88.157.45
8         39789       172.129.22.61 ->         9779
                      80.88.157.61
9         36876       80.88.157.59 ->          9775
                      172.129.22.59
10        53041       172.129.22.5 ->          9679
                      80.88.157.5
11        49010       172.129.22.47 ->         9896
                      80.88.157.47
12        53082       172.129.22.13 ->         9650
                      80.88.157.13
13        63027       80.88.157.69 ->          9617
                      172.129.22.69
14         52722       80.88.157.73 ->          10001
                      172.129.22.73
15        55165       80.88.157.71 ->          9924
```

- CPU ID – номер ядра CPU;
- CPU load – загрузка ядра CPU (pps);
- Heaviest session – сессия, которая нагружает ядро CPU больше всего;
- Session weight – утилизация сессии (pps).

show cpu history

Данной командой осуществляется просмотр истории использования ресурсов CPU.

Синтаксис

```
show cpu history [ average | max ] [cpu {<CPU>}] [timer {<TIMER>}]
```

Параметры

average | max – указывает, историю какой статистики необходимо выводить, усредненную или максимальную за интервал. Без указания данного ключа выводится история усредненной статистики (average).

<CPU> – возможно указать номер конкретного CPU, по которому будет выводиться история статистики использования ресурсов CPU. Без указания номера CPU будет выводиться информация по всем CPU, использующимися системой.

<TIMER> – в качестве параметров для данного ключа могут выступать:

- hours – отображает историю за последние 72 часа;
- minutes – отображает историю за последние 60 минут;
- seconds – отображает историю за последние 60 секунд;
- При отсутствии ключа timer выводятся 3 таблицы истории статистики использования CPU.

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# show cpu history max cpu 0
CPU0
Last 60 seconds:
utilization, %
100
 90    #
 80    #
 70    #
 60    #
 50    ##
 40    ##
 30   # ##
 20   # ##
 10 ############
  ....|....|..
 0 25 50
  time, 5 sec.
Last 60 minutes:
utilization, %
100             #
 90             #
 80             #
 70 #            #
 60 ##           #
 50 ##           #
 40 ##           #
 30 ## #    #    #    #    #    #
 20 ## #    #    #    #    #    #
 10 ############################################################
  ....|....|....|....|....|....|....|....|....|....|....|....|
 0  5 10 15 20 25 30 35 40 45 50 55 60
                          time, min.
```

```
Last 72 hours:
utilization, %
100 #       # #           ##
 90 #       # ###           ##
 80 #       # ###           ##
 70 #       #####           ##
 60 #       #####           ##
 50 #       #####           ##
 40 #       #####           ##
 30 ##############################################
 20 ##############################################
 10 ##############################################
  ....|....|....|....|....|....|....|....|....|....|....|....|....|..
 0  5 10 15 20 25 30 35 40 45 50 55 60 65 70
                                time, hours
```

show cpu processes

Данной командой осуществляется просмотр использования ресурсов CPU-процессами.

Синтаксис

```
show cpu processes [ active | name <PROCESS_NAME> ]
```

Параметры

active – список активных процессов;

name – имя конкретного процесса.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show cpu processes
PID     Name                      CPU 5s    CPU 1m    CPU 5m    Memory     Runtime
                                                                          (d,h:m:s)
-----   ----------------------   -------   -------   -------   -------   -------------
1384    Routing                   0.00%     0.13%     0.13%     0.32%     00,00:45:46
 474    Modem-cfgmgr              0.00%     0.00%     0.00%     0.30%     00,00:00:56
 724    ESRFS                     0.00%     0.02%     0.00%     0.10%     00,00:00:57
 447    File-mgr                  0.00%     0.00%     0.00%     0.72%     00,00:02:51
 620    Modem-mgr                 0.00%     0.00%     0.00%     0.32%     00,00:00:18
 472    IPC-mgr                   0.00%     0.00%     0.00%     0.08%     00,00:00:45
 468    E1D                       0.00%     0.00%     0.00%     0.12%     00,00:00:31
 305    SYSLOG                    0.00%     0.00%     0.01%     0.23%     00,00:02:07
 297    HAVEGEd                   0.00%     0.00%     0.01%     0.14%     00,00:03:58
 239    Device-mgr                0.00%     0.02%     0.01%     0.11%     00,00:01:13
 319    RNGd                      0.00%     0.02%     0.02%     0.03%     00,00:07:00
 371    Service-mgr               0.00%     0.02%     0.01%     0.14%     00,00:03:45
 363    Cp-mgr                    0.00%     0.00%     0.00%     0.13%     00,00:01:25
```

```
 366    Session-mgr                0.00%     0.00%     0.00%     0.14%     00,00:01:05
 373    REXd                       0.00%     0.00%     0.00%     0.17%     00,00:00:53
 367    Lb                         0.00%     0.00%     0.00%     0.07%     00,00:00:18
 368    Alarm-mgr                  0.00%     0.00%     0.00%     0.09%     00,00:00:43
 359    Syslog-mgr                 0.00%     0.00%     0.00%     0.16%     00,00:01:25
 358    Ipc-hub                    0.00%     0.13%     0.14%     0.23%     00,00:51:46
 372    AAA-mgr                    0.00%     0.00%     0.00%     0.23%     00,00:00:42
 370    Systemdb                   0.00%     0.02%     0.03%     0.15%     00,00:09:11
 374    SFP-mgr                    0.00%     0.02%     0.00%     0.22%     00,00:01:20
 362    If-mgr-ng                  0.00%     0.00%     0.00%     0.27%     00,00:01:33
 369    Env-mgr                    0.00%     0.03%     0.04%     0.23%     00,00:15:00
 360    Lic-mgr                    0.00%     0.03%     0.04%     0.24%     00,00:14:03
 361    If-mgr                     0.80%     0.60%     0.65%     0.36%     00,03:54:54
 365    Cfgsync-mgr                0.00%     0.07%     0.02%    10.85%     00,00:37:08
 364    Oi-mgr                     0.20%     0.08%     0.08%     3.09%     00,00:31:08
31093   CLI                        0.00%     0.10%     0.02%     0.58%     00,00:00:01
```

show cpu utilization

Данной командой осуществляется просмотр использования ресурсов CPU.

Синтаксис

```
show cpu utilization [ unit { <RANGE-ID> | local } ]
```

Параметры

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения использования ресурсов CPU устройства, на котором вводится команда, в формате вывода без включенного кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cpu utilization
CPU   Last      Last      Last
      5 sec     1 min     5 min
---   -------   -------   -------
0     1.98%     6.75%     20.02%
1     67.50%    15.62%    6.88%
2     65.43%    15.53%    6.94%
3     69.29%    16.08%    7.08%
4     89.90%    20.79%    9.14%
5     74.95%    17.14%    7.49%
6     87.61%    20.18%    8.85%
7     87.41%    20.17%    8.85%
8     81.84%    19.03%    8.40%
9     84.82%    19.79%    8.73%
10    84.53%    19.78%    8.75%
```

```
11     83.02%     19.40%     8.58%
12     83.73%     19.55%     8.63%
13     76.56%     16.99%     7.25%
14     70.47%     16.00%     6.95%
15     68.39%     15.07%     6.40%
```

show licence-manager status

Данной командой осуществляется просмотр информации о взаимодействии с сервером ELM.

Синтаксис

```
show licence-manager status
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# show licence-manager status
ELM server type:                root
Last request status:            success
Last request to licence server: 2024-08-12 08:57:07
Next request to licence server: 2024-08-12 09:57:07
```

show system

Данной командой осуществляется просмотр параметров окружения устройства.

Синтаксис

```
show system
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show system
System type:                Rusteletech R800 Service Router
System name:                rtt
Software version:           1.24.0 build 1[d9bdbda] (2024-08-01 18:06:41)
Hardware version:           1v3
System uptime (d,h:m:s):    00,00:04:05
System MAC address:         A8:F9:4B:AA:03:A0
System serial number:       NP01000050
Main power supply installed:     Present
Main power supply status:        Ok
Reserve power supply installed:  Absent
Fan Level:                        46%
   Fan Table
   ~~~~~~~~~
         Fan 1   Fan 2   Fan 3   Fan 4
------   -----   -----   -----   -----
Status   Ok      Ok      Ok      Ok
   Temperature Table
   ~~~~~~~~~~~~~~~~~
                 CPU         Sensor 1   Sensor 2   Sensor 3
--------------   --------    --------   --------   --------
Temperature, C   63          39         37         49
   Memory Table
   ~~~~~~~~~~~~
        Total, MB           Used, MB            Free, MB
-----   -----------------   -----------------   -----------------
RAM     3798.25             1643.50 (44%)       2154.75 (56%)
FLASH   20.00               1.06 (6%)           18.94 (94%)
```

show system id

Данной командой осуществляется просмотр серийного номера устройства.

Синтаксис

```
show system id
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show system id
Serial number:
```

show unit id

Данной командой просматривается номер текущего юнита устройства, а также номер юнита устройства, который будет применен после перезагрузки.

Синтаксис

```
show unit id
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show unit id
Unit ID is 1
Unit ID will be 1 after reboot
```

system config-confirm timeout

Данная команда определяет интервал времени ожидания для подтверждения текущей конфигурации. Если конфигурация не будет подтверждена, то после истечения времени ожидания произойдет откат на предыдущую примененную конфигурацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
system config-confirm timeout <TIME>
no system config-confirm timeout
```

Параметры

<TIME> – интервал времени ожидания подтверждения резервирования конфигурации, принимает значение в секундах [120..86400].

Значение по умолчанию

600

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system config-confirm timeout 1200
```

system cpu load-balance overload-threshold

Данная команда определяет пороговое значение нагрузки ядра CPU (CPU load), при котором запускается механизм перебалансировки сессий между ядрами CPU.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
system cpu load-balance overload-threshold <CPU-LOAD>
no system cpu load-balance overload-threshold
```

Параметры

<CPU_LOAD> – значение порога срабатывания перебалансировки [1000-200000].

Значение по умолчанию

30000 – для R100 и R800;

60000 – R200.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system cpu load-balance overload-threshold 15000
```

system fan-speed

Команда определяет режим работы системы охлаждения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
system fan-speed { auto | max }
no system fan-speed
```

Параметры

auto – режим автоматического регулирования;

max – режим максимального охлаждения.

Значение по умолчанию

auto – для R100/200;

max – для R800.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system fan-speed auto
```

system prompt

Команда изменяет строку приглашения в CLI устройства, позволяя отобразить заданную информацию в указанном формате. Изменения применяются после повторного входа в CLI. Формат задается с использованием "prompt" или 'prompt', а информационные ключи заключаются в %key%.

Использование отрицательной формы команды (no) устанавливает строку приглашения по умолчанию.

Синтаксис

```
system prompt <PROMPT>
no system prompt
```

Параметры

<PROMPT> – ключи, доступные для использования:

- \n – перенос строки;
- u – выводит номер локального юнита;
- h – выводит hostname устройства;
- r – выводит роль локального юнита в cluster;
- s – выводит статус полной синхронизации в cluster;

- v<NUM><VRF> – выводит статус vrrp id, где <NUM> – номер vrrp id; <VRF> – наименование VRF, если присутствует;
- vg<NUM><VRF> – выводит статус vrrp group, где <NUM> – номер vrrp group; <VRF> – наименование VRF, если присутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system prompt 'Cluster: %s% | Unit: %u% | State: %r%\nVRRP:\nVRRP id 1: %v1%'
rtt(config)# end
rtt(config)# commit
rtt(config)# confirm
Configuration has been confirmed. Commit timer canceled.
rtt# exit
rtt login: admin
rtt login: password

****************************************
*            Welcome to RTT            *
****************************************

Cluster: Yes | Unit: 1 | State: Active
VRRP:
VRRP id 1: Master#
```

system utilization softgre disable

Данная команда используется для отключения подсчета загруженности SoftGRE-туннелей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] system utilization softgre disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Подсчет включен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system utilization softgre disable
```

update licence-manager licence

Данная команда позволяет принудительно отправить запрос на сервер ELM для получения актуальной информации о лицензии, не дожидаясь таймеров.

Синтаксис

```
update licence-manager licence
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# update licence-manager licence
```

9.2. Настройка подключения к серверу лицензирования

descriprtion

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
[no] description <DESCRIPTION>
```

Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# description protected-server
```

enable

Данной командой включается менеджер лицензирования.

Использование отрицательной формы команды (no) отключает его.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# enable
```

host address

Данной командой задаётся адрес сервера лицензирования.

Синтаксис

```
host address { <ADDR> | <IPV6-ADDR> | <WORD> }
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<WORD> – DNS-имя сервера, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# host address elm.Rusteletech.ru
```

host port

Данной командой задаётся номер TCP-порта сервера лицензирования.

Использование отрицательной формы команды (no) удаляет номер TCP-порта.

Синтаксис

```
host port <PORT>
no host port
```

Параметры

<PORT> – номер TCP-порта, принимает значения [1..65535].

Значение по умолчанию

8099

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# host port 8098
```

ip vrf forwarding

Данной командой задаётся имя экземпляра VRF, в котором будет работать менеджер лицензирования.

Использование отрицательной формы команды (no) удаляет привязку к экземпляру VRF.

Синтаксис

```
[no] ip vrf forwarding <VRF>
```

Параметры

<VRF> – имя VRF, задаётся строкой до 31символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
vrtt(config-licence-manager)# ip vrf forwarding LOCAL_VRF
```

licence-key

Данной командой задаётся лицензионный ключ vRTT для аутентификации на сервере ELM.

Использование отрицательной формы команды (no) удаляет этот ключ.

Данный параметр необходимо указывать только на устройстве vRTT.

Синтаксис

```
licence-key <KEY>
no licence-key
```

Параметры

<KEY> – лицензионный ключ, задаётся строкой до 128 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
vrtt(config-licence-manager)# licence-key 1BC29B36F623BA82AAF6724FD3B16718
```

licence-manager

Данной командой осуществляется переход в режим конфигурирования менеджера лицензирования.

Синтаксис

```
licence-manager
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# licence-manager
```

location

Данной командой задаётся текстовое описание, которое передаётся на сервер ELM.

Использование отрицательной формы команды (no) удаляет это описание.

Синтаксис

```
location <WORD>
no location
```

Параметры

<WORD> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# location "Server room in Moscow office <3"
```

show licence

Данная команда служит для просмотра информации об активных лицензиях устройства.

Feature – наименование параметра лицензии;

Source – источник, из которого была получена лицензия:

- **ELM** – лицензия была получена от Eltex Licence Manager;
- **File** – лицензия была загружена из файла;
- **Boot** – лицензия была установлена на производстве Eltex.

State – состояние лицензии:

- **Active** – параметр лицензии применён;
- **Candidate** – параметр лицензии будет применён после перезагрузки устройства;
- **Unsupported** – параметр лицензии не поддержан на этой версии ПО.

Value – значение параметра лицензии;

Valid from – время начала действия параметра лицензии;

Expiries – время окончания действия параметра лицензии.

Синтаксис

```
show licence
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# show licence
Feature                 Source     State       Value         Valid from     Expiries
--------------------   --------   --------   ------------   ------------   ------------
WIFI                   ELM        Active        true          --                  --
TEST                   ELM        Unsupported   true          --                  --
BRAS                   File       Active        true          --                  --
BRAS                   File       Candidate     true          --                  --
FSTEC-BOOT             Boot       Active        true          --                  --
FSTEC-BOOT             Boot       Candidate     true          --                  --
```

system-name

Данной командой задаётся текстовое имя устройства, которое передаётся на сервер ELM.

Использование отрицательной формы команды (no) присваивает этому имени значение по умолчанию.

Синтаксис

```
system-name <WORD>
no system-name
```

Параметры

<WORD> – имя, задаётся строкой до 253 символа.

Значение по умолчанию

По умолчанию значение system-name совпадает с hostname.

Необходимый уровень привилегий

15

Командный режим

CONFIG-LICENCE-MANAGER

Пример

```
rtt(config-licence-manager)# system-name main-office
```

10. УПРАВЛЕНИЕ СИСТЕМНЫМИ ЧАСАМИ

clock timezone

Данной командой устанавливается часовой пояс.

Использование отрицательной формы команды (no) устанавливает часовой пояс по умолчанию.

Синтаксис

```
clock timezone <OFFSET>
no clock timezone
```

Параметры

<OFFSET> – обозначение зоны, содержащее сдвиг в часах относительно Greenwich Mean Time, принимает значения [gmt -12 .. gmt +12].

Значение по умолчанию

gmt 0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# clock timezone gmt +7
```

burst

Данной командой включается отправка нескольких пакетов вместо одного, когда сервер доступен для точного измерения джиттера с большими интервалами опроса. Количество пакетов в серии определяется разницей между текущей экспонентой опроса и минимальной экспонентой опроса как степень 2 (двойки) по формуле: burst = min($2^{(hpoll-minpoll)}$, 8) - 1,

где:

min(x) – функция, выбирающая наименьшее значение из двух переменных, перечисленных через запятую в скобках;

hpoll – степень текущего интервала опроса пира;

minpoll – степень минимального интервала опроса пира.

Использование отрицательной формы команды (no) выключает отправку нескольких пакетов при установлении соединения.

Синтаксис

```
[no] burst
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-NTP

Пример

```
rtt(config-ntp)# burst
```

iburst

Данной командой включается отправка восьми пакетов вместо одного при инициализации синхронизации часов с пиром, что приводит к сокращению времени синхронизации. После достижения синхронизации число отправляемых пакетов сокращается до значения minpoll.

Эта опция считается "агрессивной" для некоторых публичных NTP-серверов, её включение может привести к ошибкам или отказу синхронизации.

Использование отрицательной формы команды (no) выключает отправку нескольких пакетов при разрыве соединения.

Синтаксис

```
[no] iburst
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-NTP

Пример

```
rtt(config-ntp)# iburst
```

key

Данной командой определяется ключ из списка доверенных ключей.

Использование отрицательной формы команды (no) удаляет привязку к указанному ключу.

Синтаксис

```
key <ID>
no key
```

Параметры

<ID> – идентификатор ключа, задается в диапазоне [1..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-NTP

Пример

```
rtt(config-ntp)# key 245
```

maxpoll

Данная команда устанавливает максимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
maxpoll <INTERVAL>
no maxpoll
```

Параметры

<INTERVAL> – максимальное значение интервала опроса. Параметр команды используется как показатель степени двойки при вычислении длительности интервала в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [4..17].

Значение по умолчанию

10 (2^{10} = 1024 секунды или 17 минут 4 секунды).

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

```
rtt(ntp-remote)# maxpoll 11
```

minpoll

Данная команда устанавливает минимальное значение интервала времени между отправкой сообщений NTP-серверу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
minpoll <INTERVAL>
no minpoll
```

Параметры

<INTERVAL> – минимальное значение интервала опроса в секундах, вычисляется путем возведения двойки в степень, заданную параметром команды, принимает значение [1..6].

Значение по умолчанию

6 (2^{6} = 64 секунды или 1 минута 4 секунды).

Необходимый уровень привилегий

10

Командный режим

CONFIG-NTP

Пример

```
rtt(ntp-remote)# minpoll 4
```

ntp access-addresses

Данной командой определяется список доверенных IP-адресов, с которыми может происходить обмен ntp-пакетами.

Использование отрицательной формы команды (no) удаляет указанный список.

Синтаксис

```
ntp access-addresses <NAME>
no ntp access-addresses
```

Параметры

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Фильтрация отключена.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
object-group network TIME
  ip prefix 10.0.0.0/25
  ipv6 prefix 2001::/64
exit

rtt(config)# ntp access-addresses TIME
```

ntp authentication enable

Данная команда включает аутентификацию для NTP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ntp authentication enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp authentication enable
```

ntp authentication key-chain

Данная команда определяет набор паролей для аутентификации через алгоритм хеширования md5 с сервером, пиром.

Использование отрицательной формы команды (no) удаляет привязку к набору паролей.

Синтаксис

```
ntp authentication key-chain <KEYCHAIN>
no ntp authentication key-chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp authentication key-chain lock
```

ntp authentication trusted-key

Данной командой определяется список доверенных ключей из набора ключей.

Использование отрицательной формы команды (no) удаляет указанный ключ.

Синтаксис

```
[no] ntp authentication trusted-key <ID>
```

Параметры

<ID> – идентификатор ключа, задается в диапазоне [1..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp authentication trusted-key 25
```

ntp broadcast-client enable

Данной командой включается режим приёма широковещательных сообщений NTP-серверов для глобальной конфигурации и всех существующих VRF. Маршрутизатор работает в качестве NTP-клиента. Если в конфигурации устройства заданы NTP-пиры и серверы, то в широковещательном режиме они игнорируются.

Использование отрицательной формы команды (no) выключает широковещательный режим.

Синтаксис

```
[no] ntp broadcast-client enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp broadcast-client enable
```

ntp dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов NTP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ntp dscp <DSCP>
no ntp dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

46

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp dscp 40
```

ntp enable

Данной командой включается синхронизация системных часов с удаленными серверами по протоколу NTP для глобальной конфигурации и всех созданных VRF.

Использование отрицательной формы команды (no) выключает синхронизацию по протоколу NTP.

Синтаксис

```
[no] ntp enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp enable
```

ntp ipv6 source address

Данная команда используется для указания source-IPv6-адреса для NTP-пакетов для всех peer.

Использование отрицательной формы команды (no) устанавливает source-IPv6-адрес для NTP-пакетов по умолчанию.

Синтаксис

```
ntp ipv6 source address <IPV6-ADDR>
no ntp ipv6 source address
```

Параметры

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Значение по умолчанию

IPv6-адрес интерфейса, с которого отправляется NTP-пакет.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp ipv6 source address fc00::1
```

ntp master

Данная команда используется для активации сервиса автономного источника точного времени.

Использование отрицательной формы команды (no) отключает ntp-сервер на устройстве.

Синтаксис

```
ntp master <STRATUM>
no ntp master
```

Параметры

<STRATUM> – параметр, который определяет точность часов, задается в виде десятичного числа в диапазоне 1–15.

Параметр не является обязательным.

Значение по умолчанию

8

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp master

rtt(config)# ntp master stratum 9
```

ntp object-group query-only

Данная команда включает режим query-only, ограничивающий взаимодействие по NTP для определенного профиля IP-адресов, входящих в эту группу. В этом режиме применяются следующие ограничения: запрещена синхронизация времени между одноранговыми пирами. Ответы на NTP-запросы (отдача времени) и обмен control messages разрешены.

Использование отрицательной формы команды (no) отключает данные ограничения.

Синтаксис

```
ntp object-group query-only <NAME>
no ntp object-group query-only
```

Параметры

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Работа с NTP-пирами ничем не ограничивается.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp object-group query-only OG-NTP-QO
```

ntp object-group serve-only

Данная команда включает режим serve-only, ограничивающий взаимодействие по NTP для определенного профиля IP-адресов, входящих в эту группу. В этом режиме применяются следующие ограничения: запрещен обмен control messages и синхронизация времени между одноранговыми пирами. Ответы на ntp-запросы (отдача времени) разрешены.

Использование отрицательной формы команды (no) отключает данные ограничения.

Синтаксис

```
ntp object-group serve-only <NAME>
no ntp object-group serve-only
```

Параметры

<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Работа с NTP-пирами ничем не ограничивается.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp object-group serve-only OG-NTP-SO
```

ntp peer

Данная команда используется для установления партнерских отношений между NTP-серверами и перехода в командный режим CONFIG-NTP-PEER.

NTP-сервер на маршрутизаторе работает в режиме двусторонней активности с удаленным NTP-сервером, указанным в команде. В случае потери связи одного из партнеров с вышестоящим NTP-сервером он сможет синхронизировать время по серверу-партнеру.

Использование отрицательной формы команды (no) удаляет заданного NTP-партнера.

Синтаксис

```
[no] ntp peer { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес партнера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес партнера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-партнер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp peer 10.100.100.1
rtt(config-ntp-peer)#
```

ntp pool

Данная команда используется для создания NTP-pool и перехода в командный режим CONFIG-NTP-POOL.

Директива pool позволяет динамически получать IP-адреса серверов из указанного пула. Когда серверы становятся недоступными, они отбрасываются, а новые серверы добавляются на лету, поэтому в любой момент можно успешно опросить определенное количество серверов.

Использование отрицательной формы команды (no) удаляет заданный NTP-pool.

Синтаксис

```
[no] ntp pool { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp pool 10.100.100.3
rtt(config-ntp-pool)#
```

ntp server

Данная команда используется для создания NTP-сервера и перехода в командный режим CONFIG-NTP-SERVER.

Маршрутизатор работает с указанным NTP-сервером в режиме односторонней активности. В данном режиме локальные часы маршрутизатора могут синхронизироваться с удаленным NTP-сервером.

Использование отрицательной формы команды (no) удаляет заданный NTP-сервер.

Синтаксис

```
[no] ntp server { <ADDR> | <NAME> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NAME> – DNS-имя сервера, задаётся строкой до 31 символа;

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать NTP-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp server 10.100.100.2
rtt(config-ntp-server)#
```

ntp source address

Данная команда используется для указания source-IP-адреса для NTP-пакетов для всех peer.

Использование отрицательной формы команды (no) устанавливает source-IP-адрес для NTP-пакетов по умолчанию.

Синтаксис

```
ntp source address <ADDR>
no ntp source address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

IP-адрес интерфейса, с которого отправляется NTP-пакет.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ntp source address 10.100.100.2
```

prefer

Команда отмечает данный NTP-сервер как предпочтительный при выборе из списка доступных (активных) NTP-серверов. Данная команда предоставляет возможность повысить приоритет при выборе NTP-сервера, но это никак не повлияет на работу текущей синхронизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] prefer
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-NTP

Пример

```
rtt(ntp-remote)# prefer
```

set date

Данной командой устанавливается вручную системное время и дата.

Синтаксис

```
set date <TIME> [<DAY> <MONTH> [ <YEAR> ] ]
```

Параметры

<TIME> – устанавливаемое системное время, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

<DAY> – день месяца, принимает значения [1..31];

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/ September/October/November/December];

<YEAR> – год, принимает значения [2001..2037].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# set date 16:35:00 15 May 2014
```

show date

Данная команда позволяет посмотреть текущие системное время и дату.

Синтаксис

```
show date
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show date
```

```
2024-07-01 09:14:09"
```

show ntp configuration

Данная команда отображает действующую (RUNNING) конфигурацию протокола NTP.

Синтаксис

```
show ntp configuration
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show ntp configuration
NTP status:          Enabled
NTP mode:            client/server
NTP DSCP:            46

Address       VRF instance   Type  Version  Min poll  Max poll  Prefer  burst  iburst
-------   ----------------  ----   -------  -------   --------  ------   ------   ----
0.pool.ntp.org    --         pool     4       6          10      No       Yes      No
1.pool.ntp.org    --         server   4       6          10      No       No       Yes
2.pool.ntp.org    --         peer     4       6          10      No       No       No
```

show ntp peers

Данная команда позволяет посмотреть текущее состояние NTP-серверов (пиров). В следующих таблицах приведен перечень отображаемых параметров и их описание.

Таблица 6 – Состояние удаленного сервера (пира)

Параметр	Описание
vrf	Имя экземпляра VRF.
remote	DNS-имя или IP-адрес сервера (пира). Первый символ в таблице используется для обозначения состояния сервера (пира), состояния описаны в таблице 7.
refid	Идентификатор связи или IP-адрес того, с кем синхронизирован удаленный сервер (пир). Типы идентификаторов связи описаны в таблице 8.
st	Стратум.

Параметр	Описание
t	Отношение маршрутизатора к удаленному серверу (пиру), типы описаны в таблице 9.
when	Период времени с момента, когда сервер (пир) последний раз опрашивался, в секундах (“h” часы,“d” дни).
poll	Частота опроса сервера (пира).
reach	Восьмибитный сдвигаемый влево регистр, содержащий результаты опросов (1 = успешно, 0 = неуспешно), отображается в восьмеричной системе счисления.
delay	Время прохождения пакета до сервера (пира) и обратно, в миллисекундах.
offset	Среднее постоянное смещение времени маршрутизатора относительно сервера (пира).
jitter	Средний разброс отклонения времени (джиттер).

Таблица 7 – Состояние удаленного сервера (пира)

Тип	Описание
пробел	Указывает на то, что: • не было ответов от удаленного сервера (пира); • сервер не используется, так как стратум имеет большое значение; • сервер (пир) использует данный маршрутизатор для синхронизации своих часов.
x	Сервер (пир) не используется для синхронизации времени, отброшен алгоритмом пересечения.
-	Сервер (пир) не используется для синхронизации времени, отброшен кластерным алгоритмом.
#	Рабочий удаленный сервер (пир), но не используется, так как не вошел в число первых шести серверов (пиров), отсортированных по расстоянию синхронизации, является резервным.
+	Рабочий и предпочитаемый удаленный сервер (пир), включен алгоритмом объединения.
*	Сервер (пир), который в настоящее время является первичным источником времени.

Таблица 8 – Типы идентификаторов соединения с удаленным сервером (пиром)

Тип	Описание
.ACST.	Manycast-сервер.
.AUTH.	Ошибка аутентификации.
.AUTO.	Ошибка последовательности автоматического ключа.
.BCST.	Broadcast-сервер.
.CRYPT.	Ошибка протокола автоматического ключа.
.DENY.	Сервер отказал в доступе.
.INIT.	Инициализация соединения с сервером.
.MCST.	Multicast-сервер.
.TIME.	Таймаут соединения с сервером.

Тип	Описание
.STEP.	Ступенчатое изменение времени, смещение меньше предельного порога (1000 миллисекунд), но больше, чем шаг порога (125 миллисекунд).
.RATE.	Превышение частоты опросов.

Таблица 9 – Типы отношений маршрутизатора к удаленному серверу (пиру)

Тип	Описание
u	Unicast или manycast-клиент.
b	Broadcast или unicast-клиент.
p	Директива pool.
s	Двусторонняя связь (пир).
A	Manycast-сервер.
B	Broadcast-сервер.
M	Multicast-сервер.

Синтаксис

```
show ntp peers
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh ntp peers
 Clock is synchronized, stratum 4, reference is 162.159.200.123
     remote                                          vrf          refid            st   t   when   poll   reach   delay      offset     jitter
 -   ------------------------------------------------   ----------   ---------------  --   -   ----   ----   -----   --------   --------   --------
 *   162.159.200.123                                                 10.87.12.180     3    u   0      8      15      43.681     39.153     0.608
 +   91.207.136.55                                                   89.109.251.22    2    u   2      8      15      61.094     42.103     0.182
 +   162.159.200.1                                                   10.87.12.180     3    u   3      8      15      51.740     35.247     4.176
     0.ru.pool.ntp.org                                               POOL             16   p   0      8      0       0.000      0.000      0.001
```

version

Данной командой устанавливается версия NTP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
version <VERSION>
no version
```

Параметры

<VERSION> – версия NTP-протокола, принимает значения [1..4].

Значение по умолчанию

4

Необходимый уровень привилегий

15

Командный режим

CONFIG-NTP

Пример

```
rtt(ntp-remote)# version 3
```

11. НАСТРОЙКА AAA

aaa accounting commands

Данной командой конфигурируется список способов учета команд, введённых в CLI.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
aaa accounting commands stop-only <METHOD>
no aaa accounting commands stop-only
```

Параметры

<METHOD> – способы учета:

- tacacs – учет введенных команд по протоколу TACACS.

Значение по умолчанию

Учёт не ведется.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa accounting commands stop-only tacacs
```

aaa accounting login

Данной командой конфигурируется список способов учета сессий пользователей. Ведение учета активируется и прекращается, когда пользователь входит и отключается от системы, что соответствует значениям «start» и «stop» в сообщениях протоколов RADIUS и TACACS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
aaa accounting login start-stop <METHOD 1> [ <METHOD 2> ]
no aaa accounting login start-stop
```

Параметры

<METHOD> – способы учета:

- tacacs – учет сессий по протоколу TACACS;
- radius – учет сессий по протоколу RADIUS.

Значение по умолчанию

Учет сессий ведется в локальный журнал.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa accounting login start-stop tacacs
```

aaa authentication attempts max-fail

Данной командой устанавливается максимальное количество неудачных попыток аутентификации до блокировки пользователя и время, на которое происходит блокировка.

Использование отрицательной формы команды (no), значения количества попыток и период блокировки устанавливает по умолчанию.

Синтаксис

```
aaa authentication attempts max-fail <COUNT> <TIME>
no aaa authentication attempts max-fail
```

Параметры

<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535];

<TIME> – интервал времени в секундах, на который будет заблокирован пользователь, принимает значения [1..65535].

Значение по умолчанию

Количество неудачных попыток – 5.

Период блокировки – 300.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authentication attempts max-fail 5 30
```

aaa authentication enable

Данной командой создаются списки способов аутентификации повышения привилегий пользователей. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему способу в списке.

В конфигурации по умолчанию существует список с именем «default». Список «default» содержит один способ аутентификации – «enable». Чтобы использовать список для аутентификации повышения привилегий пользователей, необходимо выполнить его привязку командой, описанной в разделе enable authentication.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

Синтаксис

```
aaa authentication enable <NAME> <METHOD 1> [ <METHOD 2> ] [ <METHOD 3> ] [ <METHOD 4> ]
no aaa authentication enable <NAME>
```

Параметры

<NAME> – имя списка: строка до 31 символа;

- default – имя списка «default».

<METHOD> – способы аутентификации:

- enable – аутентификация с помощью enable-паролей;
- tacacs – аутентификация по протоколу TACACS;
- radius – аутентификация по протоколу RADIUS;
- ldap – аутентификация по протоколу LDAP.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authentication enable enable-test tacacs enable
```

aaa authentication login

Данной командой создаются списки способов аутентификации входа пользователей в систему. При неудачной попытке аутентификации по одному способу происходит попытка аутентификации по следующему в списке.

В конфигурации по умолчанию существует список с именем «default», данный список содержит один способ аутентификации – «local». Чтобы использовать список для аутентификации входа пользователей, необходимо выполнить его активацию командой, описанной в разделе login authentication.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

Синтаксис

```
aaa authentication login { default | <NAME> } <METHOD 1> [ <METHOD 2> ] [ <METHOD
3> ]
[ <METHOD 4> ]
no aaa authentication login { default | <NAME> }
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Способы аутентификации:

- local – аутентификация с помощью локальной базы пользователей;
- tacacs – аутентификация по списку TACACS-серверов;
- radius – аутентификация по списку RADIUS-серверов;
- ldap – аутентификация по списку LDAP-серверов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authentication login login-test tacacs local
```

aaa authentication mode

Данной командой определяется режим работы со списками методов аутентификации.

Использование отрицательной формы команды (no) удаляет список способов аутентификации.

Синтаксис

```
[no] aaa authentication mode { break | chain }
```

Параметры

break – при аутентификации будут использоваться последующие методы в случае недоступности более приоритетного;

chain – при аутентификации будут использоваться последующие методы в случае получения отказа от более приоритетного.

Значение по умолчанию

chain

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authentication mode break
```

aaa authorization commands

Данной командой создаются списки способов авторизации команд, вводимых пользователем в систему. При неудачной попытке авторизации по одному способу происходит попытка аутентификации по следующему способу в списке.

В конфигурации по умолчанию существует список с именем «default», данный список содержит один способ авторизации – «local». Чтобы использовать список для авторизации команд, вводимых пользователем в систему, необходимо выполнить его активацию командой, описанной в разделе commands authorization.

Использование отрицательной формы команды (no) удаляет список способов авторизации.

Синтаксис

```
aaa authorization commands { default | <NAME> } <METHOD 1> [ <METHOD 2> ]
no aaa authorization commands { default | <NAME> }
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Способы аутентификации:

- local – авторизация с помощью локальной базы пользователей;
- tacacs – авторизация по списку TACACS-серверов;

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authorization commands
```

aaa authorization control-commands enable

Данной командой включается авторизация контрольных команд (help, logout, end, exit) на TACACS-сервере. В конфигурации по умолчанию эта функция отключена.

Использование отрицательной формы команды (no) возвращает состояние к дефолтному.

Синтаксис

```
[ no ] aaa authorization control-commands enable
```

Значение по умолчанию

no aaa authorization control-commands enable

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authorization control-commands enable
```

aaa authorization mode

Данной командой определяется режим работы со списками методов авторизации.

Использование отрицательной формы команды (no) удаляет список способов авторизации.

Синтаксис

```
[no] aaa authorization mode { break | chain }
```

Параметры

break – при аутентификации будут использоваться последующие методы в случае недоступности более приоритетного;

chain – при аутентификации будут использоваться последующие методы в случае получения отказа от более приоритетного.

Значение по умолчанию

chain

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authorization mode break
```

aaa authorization remote user privilege

Данной командой устанавливается уровень привилегий, назначаемых пользователям, авторизованным через AAA-сервера, но не получившим от них уровень привилегий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
aaa authorization remote user privilege <PRIV>
no aaa authorization remote user privilege
```

Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [1..15].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa authorization remote user privilege 15
```

aaa das-profile

Данная команда используется для добавления профиля серверов динамической авторизации (DAS) и перехода в командный режим DAS SERVER PROFILE.

Использование отрицательной формы команды (no) удаляет заданный профиль серверов динамической авторизации (DAS).

Синтаксис

```
[no] aaa das-profile <NAME>
```

Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa das-profile profile1
rtt(config-aaa-das-profile)#
```

aaa disable

Данной командой отключает доступ на маршрутизатор через консольный интерфейс.

При использовании отрицательной формы команды (no) доступ на маршрутизатор через консольный интерфейс включается.

Синтаксис

```
[no] aaa disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Доступ на маршрутизатор через консольный интерфейс включен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LINE-CONSOLE

Пример

```
rtt(config-line-console)# aaa disable
```

aaa radius-profile

Данная команда используется для добавления профиля RADIUS-серверов и перехода в командный режим RADIUS SERVER PROFILE.

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

Синтаксис

```
[no] aaa radius-profile <NAME>
```

Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# aaa radius-profile profile1
rtt(config-aaa-radius-profile)#
```

acct-port

Данной командой задаётся номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аккаунтинга.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
acct-port <PORT>
no acct-port
```

Параметры

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

Значение по умолчанию

1813

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config-radius-server)# acct-port 4444
```

admin login enable

Данной командой включается возможность авторизации для предустановленного пользователя "admin".

Использование отрицательной формы команды (no) отключает возможность авторизации пользователя "admin".

Синтаксис

```
[no] admin login enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Команда присутвует в конфигурации по умолчанию, для пользователя "admin" разрешена авторизация.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# no admin login enable
```

auth-port

Данной командой задаётся номер порта для обмена данными с удаленным RADIUS-сервером при выполнении аутентификации и авторизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
auth-port <PORT>
no auth-port
```

Параметры

<PORT> – номер UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

Значение по умолчанию

1812

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config-radius-server)# auth-port 4444
```

clear users blocked

Данной командой удаляется информация о неправильных попытках аутентификации различных пользователей.

Синтаксис

```
clear users blocked <NAME>
```

Параметры

<NAME> – имя пользователя, для которого необходимо очистить статистику неправильных попытках аутентификации, задаётся строкой до 31 символа.

Без указания имени пользователя очищается вся таблица неправильных попыток аутентификации.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear users blocked
```

clear user-session

Данной командой закрывается рабочая сессия пользователя CLI.

Синтаксис

```
clear user-session [ <USERNAME> | <SESSION> ]
```

Параметры

<NAME> – имя пользователя сессию которого необходимо закрыть, задаётся строкой до 31 символа;

<SESSION> – номер терминальной сессии которую необходимо закрыть, задаётся число в диапазоне [1..10];

Без указания параметров – будут закрыты все активные сессии.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear users-session
```

clients

Данной командой определяется список клиентов динамической авторизации (DAC), на запросы которых будет отвечать сервер динамической авторизации (DAS).

Использование отрицательной формы команды (no) удаляет список клиентов динамической авторизации (DAC).

Синтаксис

```
clients object-group <NAME>
no clients
```

Параметры

<NAME> – имя профиля IP-адресов, содержащий адреса клиентов динамической авторизации, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-DAS-SERVER

Пример

```
rtt(config-das-server)# clients object-group pcrf
```

commands authorization

Данной командой осуществляется активация списка авторизации команд, вводимых пользователем в систему.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ авторизации – «local».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

Синтаксис

```
commands authorization <NAME>
no commands authorization
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Пример

```
rtt(config-line-ssh)# commands authorization authorization-test
```

das-server

Данная команда используется для добавления сервера динамической авторизации (DAS) и перехода в командный режим DAS SERVER. Сервера динамической авторизации (DAS) принимают RADIUS CoA запросы от клиентов динамической авторизации (DAC), например, отключение или повторный запрос списка сервисов пользователя.

Использование отрицательной формы команды (no) удаляет заданный сервер динамической авторизации (DAS).

Синтаксис

[no] das-server <NAME>

Параметры

<NAME> – имя сервера динамической авторизации (DAS), задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# das-server main
rtt(config-das-server)#
```

das-server

Данная команда используется для добавления сервера динамической авторизации (DAS) в конфигурируемый профиль серверов динамической авторизации.

Использование отрицательной формы команды (no) удаляет заданный сервер динамической авторизации (DAS).

Синтаксис

```
[no] das-server <NAME>
```

Параметры

<NAME> – имя сервера динамической авторизации (DAS), задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-AAA-DAS-PROFILE

Пример

```
rtt(config)# das-server mainrtt(config-das-server)#
```

dead-interval

Данной командой задаётся интервал, в течение которого на RADIUS-сервер не будут отправляться пакеты. В данное состояние RADIUS-сервер переводится по истечении таймаута ожидания ответа на запрос последнего допустимого повтора (см. раздел radius-server retransmit).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
dead-interval <SEC>
no dead-interval
```

Параметры

<SEC> – период времени в секундах, принимает значения [0..3600].

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config-radius-server)# dead-interval 600
```

description

Команда используется для изменения описания профиля серверов динамической авторизации (DAS) или профиля RADIUS-серверов.

Использование отрицательной формы команды (no) удаляет описание профиля.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DAS-SERVER-PROFILE

CONFIG-RADIUS-SERVER-PROFILE

Пример

Установить описание для профиля IP-адресов:

```
rtt(config-aaa-das-profile)# description "Main profile"
```

description

Команда используется для изменения описания конфигурируемого AAA-сервера (RADIUS, TACACS, LDAP).

Использование отрицательной формы команды (no) удаляет описание AAA-сервера.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание AAA-сервера, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RADIUS-SERVER

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

Пример

Установить описание для AAA-сервера:

```
rtt(config-radius-server)# description "Main server"
```

disable

Данной командой производится понижение уровня привилегий пользователя до первоначальных.

Синтаксис

```
disable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

2

Командный режим

ROOT

Пример

```
rtt# disable
rtt>
```

enable

Данной командой производится повышение уровня привилегий пользователя. Способы аутентификации повышения привилегий пользователей задаются с помощью команды, описанной в разделе aaa authentication attempts max-fail.

По умолчанию в конфигурации установлен метод аутентификации по паролю «enable». При этом пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

Для аутентификации повышения привилегий по протоколам TACACS/RADIUS/LDAP на сервере должны быть созданы пользователи $enab<PRIV>$, где <PRIV> – необходимый уровень привилегий пользователя, который должен быть аутентифицирован.

Синтаксис

```
enable [ <PRIV> ]
```

Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [2..15].

Значение по умолчанию

15

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt> enable 10
rtt#
```

enable authentication

Данной командой осуществляется активация списка аутентификации повышения привилегий пользователей, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «enable».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

Синтаксис

```
enable authentication <NAME>
no enable authentication
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Пример

```
rtt(config-line-console)# enable authentication enable-test
```

enable password

Данной командой устанавливается пароль, который будет запрашиваться при повышении уровня привилегий пользователя.

По умолчанию в конфигурации пароли не заданы, то есть любой системный пользователь может получить 15 необходимый уровень привилегий.

Использование отрицательной формы команды (no) удаляет пароль из системы.

Синтаксис

```
enable password { <CLEAR-TEXT> | encrypted <HASH_SHA512> } [ privilege <PRIV> ]
no enable password [ privilege <PRIV> ]
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 32] символов, принимает значения [0-9a-fA-F];

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов;

<PRIV> – необходимый уровень привилегий, принимает значение [2..15], значение по умолчанию 15.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# enable password 12345678 privilege 10
```

exec-timeout

Данной командой задаётся интервал, по истечении которого будет разрываться бездействующая сессия.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
exec-timeout <SEC>
no exec-timeout
```

Параметры

<SEC> – период времени в минутах, принимает значения [1..65535].

Значение по умолчанию

30 минут

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-SSH

CONFIG-LINE-TELNET

Пример

```
rtt(config-line-ssh)# exec-timeout 600
```

key

Данной командой задаётся пароль для аутентификации на удаленном сервере.

Использование отрицательной формы команды (no) удаляет заданный пароль для аутентификации на удаленном сервере.

Синтаксис

```
key ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> }
no key
```

Параметры

<TEXT> – строка [8..16] ASCII-символов (для TACACS-сервера – [1..16] ASCII-символов);

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов (для TACACS-сервера – до 120 символов).

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-DAS-SERVER

Пример

```
rtt(config-tacacs-server)# key ascii-text 12345678
```

ldap-server base-dn

Данной командой задаётся базовый DN (Distinguished name), который будет использоваться при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный базовый DN.

Синтаксис

```
ldap-server base-dn <NAME>
no ldap-server base-dn
```

Параметры

<NAME> – базовый DN, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server base-dn “dc=example,dc=com”
```

ldap-server bind authenticate root-dn

Данной командой задаётся DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный DN пользователя.

Синтаксис

```
ldap-server bind authenticate root-dn <NAME>
no bind authenticate root-dn
```

Параметры

<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server bind authenticate root-dn “cn=admin,dc=example,dc=com”
```

ldap-server bind authenticate root-password

Данной командой задаётся пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.

Использование отрицательной формы команды (no) удаляет заданный пароль пользователя.

Синтаксис

```
ldap-server bind authenticate root-password ascii-text { <TEXT> | encrypted
<ENCRYPTED-TEXT> }
no bind authenticate root-password
```

Параметры

<TEXT> – строка [8..16] ASCII-символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server bind authenticate root-password ascii-text 12345678
```

ldap-server bind timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server bind timeout <SEC>
no ldap-server bind timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server bind timeout 5
```

ldap-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ldap-server dscp <DSCP>
no ldap-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

63

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server dscp 40
```

ldap-server host

Данная команда используется для добавления LDAP-сервера в список используемых серверов и перехода в командный режим LDAP SERVER.

Использование отрицательной формы команды (no) удаляет заданный LDAP-сервер.

Синтаксис

```
[no] ldap-server host { <ADDR> | <IPV6-ADDR> | <LDAP-HOST-NAME> } [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес LDAP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес LDAP-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<LDAP-HOST-NAME> – DNS-имя LDAP-сервера, задаётся строкой до 31 символа. Для корректного сопоставления DNS-имени хоста с IP-адресом на маршрутизаторе должна быть запущена и настроена функция domain lookup enable.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server host 10.100.100.1
rtt(config-ldap-server)#
```

ldap-server naming-attribute

Данной командой задаётся имя атрибута объекта, со значением которого идет сравнение имени искомого пользователя на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server naming-attribute <NAME>
no ldap-server naming-attribute
```

Параметры

<NAME> – имя атрибута объекта, задаётся строкой до 127 символов.

Значение по умолчанию

uid

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server naming-attribute displayName
```

ldap-server privilege-level-attribute

Данной командой задаётся имя атрибута объекта, значение которого будет определять начальные привилегии пользователя на устройстве. Атрибут должен принимать значения [1..15]. Если указанный атрибут отсутствует или содержит недопустимое значение, то начальные привилегии пользователя будут соответствовать привилегиям пользователя «remote».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server privilege-level-attribute <NAME>
no ldap-server privilege-level-attribute
```

Параметры

<NAME> – имя атрибута объекта, задаётся строкой до 127 символов.

Значение по умолчанию

priv-lvl

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server privilege-level-attribute title
```

ldap-server search filter user-object-class

Данной командой задаётся имя класса объектов, среди которых необходимо выполнять поиск пользователей на LDAP-сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server search filter user-object-class <NAME>
no ldap-server search filter user-object-class
```

Параметры

<NAME> – имя класса объектов, задаётся строкой до 127 символов.

Значение по умолчанию

posixAccount

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server search filter user-object-class shadowAccount
```

ldap-server search scope

Данной командой задаётся область поиска пользователей в дереве LDAP-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server search scope <SCOPE>
no ldap-server search scope
```

Параметры

<SCOPE> – область поиска пользователей на LDAP-сервере, принимает следующие значения:

- onelevel – выполнять поиск в объектах на следующем уровне после базового DN в дереве LDAP-сервера;
- subtree – выполнять поиск во всех объектах поддерева базового DN в дереве LDAP-сервера.

Значение по умолчанию

subtree

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server search scope onelevel
```

ldap-server search timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что LDAP-сервер не нашел записей пользователей, подходящих под условие поиска.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ldap-server search timeout <SEC>
no ldap-server search timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [0..30].

Значение по умолчанию

0 – устройство ожидает завершения поиска и получения ответа от LDAP-сервера.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server search timeout 10
```

ldap-server ssl crypto

Данной командой указывается сертификат, который будет использоваться для установления SSL-соединения с LDAP-сервером.

Использование отрицательной формы команды (no) удаляет из конфигурации сертификат для установления SSL-соединения с LDAP-сервером.

Синтаксис

```
ldap-server ssl crypto <FILE-NAME>
no ldap-server ssl crypto
```

Параметры

Отсутствуют.

Значение по умолчанию

Сертификат не указан.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server ssl crypto ldap-ssl.crt
```

ldap-server ssl check-peer disable

Данной командой отключается верификация LDAP-сервера по сертификату.

Использование отрицательной формы команды (no) включает верификацию LDAP-сервера по сертификату.

Синтаксис

```
[no] ldap-server ssl check-peer disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server ssl check-peer disable
```

ldap-server ssl enable

Данной командой включается использование SSL-соединения для LDAP-подключения (LDAP over SSL).

Использование отрицательной формы команды (no) отключается использование SSL-соединения для LDAP-подключения (LDAP over SSL).

Синтаксис

```
[no] ldap-server ssl enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ldap-server ssl enable
```

line

Данной командой осуществляется переход в режим конфигурирования соответствующего терминала: локальная консоль, удаленная консоль (Telnet), удаленная защищенная консоль (SSH).

Использование отрицательной формы команды (no) устанавливает параметры терминала по умолчанию. Параметры по умолчанию описаны в разделах login authentication и enable authentication.

Синтаксис

```
[no] line <TYPE>
```

Параметры

<TYPE> – тип консоли:

- console – локальная консоль;
- telnet – удаленная консоль;
- ssh – защищенная удаленная консоль.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# line console
rtt(config-line-console)#
```

login authentication

Данной командой осуществляется активация списка аутентификации входа пользователей в систему, который будет использоваться в конфигурируемом терминале.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «local».

Использование отрицательной формы команды (no) делает список с именем «default» активным.

Синтаксис

```
login authentication <NAME>
no login authentication
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-LINE-CONSOLE

CONFIG-LINE-TELNET

CONFIG-LINE-SSH

Пример

```
rtt(config-line-console)# login authentication login-test
```

mode

Данной командой задаётся режим работы локальной учетной записи.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы локальной учетной записи, принимает следующие значения:

- cli – режим работы по умолчанию, пользователь получает доступ к интерфейсу командной строки, предназначенному для управления, просмотра состояния и мониторинга устройства;
- techsupport – пользователь получает доступ к командной оболочке, в которой выполняется процедура отладки устройства совместно с специалистами технической поддержки;
- sftp – пользователь используется для организации доступа к встроенному SFTP-серверу, возможность работы в какой-либо командой оболочке при этом у пользователя отсутствует.

Значение по умолчанию

cli

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

```
rtt(config-user)# mode cli
```

password

Команда для установки пароля определенному пользователю для входа в систему. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль пользователя из системы.

Синтаксис

```
password { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 32] символов, принимает значения [0-9a-fA-F];

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

CHANGE-EXPIRED-PASSWORD

Пример

```
rtt(config-user) password test
```

port

Данной командой задаётся номер порта для обмена данными с удаленным сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер TCP/UDP-порта для обмена данными с удаленным сервером, принимает значения [1..65535].

Значение по умолчанию

49 – для TACACS-сервера;

389 – для LDAP-сервера;

Не установлено – для DAS-сервера.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

CONFIG-DAS-SERVER

Пример

```
rtt(config-tacacs-server)# port 4444
```

priority

Данной командой задаётся приоритет использования удаленного сервера. Чем ниже значение, тем приоритетнее сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
priority <PRIORITY>
no priority
```

Параметры

<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-TACACS-SERVER

CONFIG-RADIUS-SERVER

CONFIG-LDAP-SERVER

Пример

```
rtt(config-tacacs-server)# priority 5
```

privilege

Данной командой производится установка уровня привилегий пользователя. Набор команд, который доступен пользователю, зависит от уровня привилегий. Пользователям с уровнями привилегий от 1 до 9 доступен только просмотр информации. Пользователям с уровнем привилегий от 10 до 15 доступна большая часть команд конфигурирования. Пользователям с уровнем привилегий 15 доступен полный набор команд. Требуемый необходимый уровень привилегий команд может быть изменен, описание в разделе description.

Использование отрицательной формы команды (no) устанавливает необходимый уровень привилегий по умолчанию.

Назначение начального уровня привилегий пользователям происходит следующим образом:

- необходимый уровень привилегий пользователям из локальной базы назначается указанной командой;
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу RADIUS, извлекается из атрибута cisco-avpair = "shell:priv-lvl=<PRIV>";
- необходимый уровень привилегий для пользователей, авторизовавшихся по протоколу TACACS, извлекается из атрибута priv-lvl=<PRIV>;
- уровень привилегий для пользователей, авторизовавшихся по протоколу LDAP, извлекается из атрибута, заданного командой *privilege-level-attribute*, описанной в разделе line, по умолчанию **priv-lvl=<PRIV>**.

Если при аутентификации пользователя через протоколы TACACS/RADIUS/LDAP не была получена вышеуказанная опция или была получена опция с некорректным значением, то пользователю будут назначены привилегии пользователя «remote», по умолчанию 1. Необходимый уровень привилегий пользователя «remote» можно изменить аналогично любому другому пользователю из локальной базы с помощью указанной команды.

Синтаксис

```
privilege <PRIV>
no privilege
```

Параметры

<PRIV> – необходимый уровень привилегий, принимает значение [1..15].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

```
rtt(config-user)# privilege 15
```

privilege

Данной командой производится установка минимального уровня привилегий пользователя, необходимого для выполнения команды из указанного поддерева команд.

Использование отрицательной формы команды (no) устанавливает необходимый уровень привилегий по умолчанию.

Синтаксис

```
privilege <COMMAND-MODE> level <PRIV> <COMMAND>
no privilege <COMMAND-MODE> <COMMAND>
```

Параметры

<COMMAND-MODE> – командный режим, может принимать значения:

- aaa-ldap-profile-configure;
- config;
- config-aaa-das-profile;
- config-aaa-radius-profile;
- config-access-profile;
- config-access-profile-user;
- config-acl;
- config-acl-rule;
- config-address-assignment-pool;
- config-archive;
- config-bgp;
- config-bgp-af;
- config-bgp-aggregate;
- config-bgp-group;
- config-bgp-group-af;
- config-bgp-group-af-vrf;
- config-bgp-listen;
- config-bgp-local-as;
- config-bgp-neighbor;
- config-bgp-neighbor-af;
- config-bgp-neighbor-af-vrf;
- config-bgp-vrf;
- config-bgp-vrf-af;
- config-bgp-vrf-aggregate;
- config-bgp-vrf-group;
- config-bgp-vrf-listen;
- config-bgp-vrf-local-as;
- config-bgp-vrf-neighbor;
- config-bridge;
- config-cellular-modem;
- config-cellular-profile;
- config-class-map;
- config-class-policy-map;
- config-cluster;
- config-cluster-unit;
- config-content-provider;

- config-crypto-sync;
- config-das-server;
- config-dhcp-server;
- config-dhcp-server-failover;
- config-dhcp-server-vendor-specific;
- config-dhcp-vendor-id;
- config-dn;
- config-dnat;
- config-dnat-pool;
- config-dnat-rule;
- config-dnat-ruleset;
- config-failover;
- config-firewall-failover;
- config-gre;
- config-http-profile;
- config-if-e1;
- config-if-gi;
- config-if-loopback;
- config-if-multilink;
- config-if-port-channel;
- config-if-qinq;
- config-if-sub;
- config-ike-gw;
- config-ike-keyring;
- config-ike-policy;
- config-ike-proposal;
- config-ip4ip4;
- config-ips;
- config-ips-auto-upgrade;
- config-ips-category;
- config-ips-category-rule;
- config-ips-category-rule-advanced;
- config-ips-content-filter;
- config-ips-policy;
- config-ips-policy-vendor;
- config-ips-policy-vendor-category;
- config-ips-upgrade-user-server;
- config-ipsec-policy;
- config-ipsec-proposal;
- config-ipsec-vpn;
- config-ipv6-dhcp-server;
- config-ipv6-ospf;
- config-ipv6-ospf-area;
- config-ipv6-ospf-vlink;
- config-ipv6-pl;
- config-ipv6-vrrp;
- config-ipv6-wan-rule;

- config-ipv6-wan-target;
- config-ipv6-wan-target-list;
- config-isis;
- config-key-chain;
- config-key-chain-key;
- config-l2tp;
- config-l2tp-server;
- config-l2tpv3;
- config-l2vpn;
- config-l2vpn-autodiscovery-bgp;
- config-l2vpn-eompls;
- config-l2vpn-pw;
- config-l2vpn-pw-class;
- config-ldap-server;
- config-ldp;
- config-ldp-af;
- config-ldp-af-if;
- config-ldp-neighbor;
- config-licence-manager;
- config-line-console;
- config-line-ssh;
- config-line-telnet;
- config-lt;
- config-mailserver;
- config-mailserver-domain;
- config-mpls;
- config-multipath-ipv6-route;
- config-multipath-route;
- config-net-policy;
- config-netflow-host;
- config-ntp;
- config-object-group-address-port;
- config-object-group-application;
- config-object-group-cf-kaspersky;
- config-object-group-content-filter;
- config-object-group-domain-name;
- config-object-group-email;
- config-object-group-mac;
- config-object-group-network;
- config-object-group-service;
- config-object-group-url;
- config-openvpn;
- config-openvpn-server;
- config-openvpn-server-user;
- config-ospf;
- config-ospf-area;
- config-ospf-vlink;

- config-pki-server;
- config-pl;
- config-policy-map;
- config-ppp-user;
- config-pppoe;
- config-pptp;
- config-pptp-server;
- config-radius-server;
- config-rip;
- config-ripng;
- config-route-map;
- config-route-map-rule;
- config-san;
- config-security-zone;
- config-security-zone-pair;
- config-security-zone-pair-rule;
- config-sflow-host;
- config-sla-responder;
- config-sla-test;
- config-snat;
- config-snat-pool;
- config-snat-rule;
- config-snat-ruleset;
- config-snmp;
- config-snmp-host;
- config-snmp-user;
- config-softgre;
- config-softgre-controller;
- config-subscriber-control;
- config-subscriber-default-service;
- config-subtunnel;
- config-syslog-destination;
- config-syslog-file;
- config-syslog-host;
- config-tacacs-server;
- config-tftp-server;
- config-track;
- config-trustpoint;
- config-user;
- config-vlan;
- config-vrf;
- config-vrrp;
- config-vti;
- config-wan-rule;
- config-wan-target;
- config-wan-target-list;
- config-wireguard;

- config-wireguard-peer;
- config-wireguard-server;
- config-wireguard-server-peer;
- config-zabbix-agent;
- config-zabbix-proxy;
- debug;
- root;
- root-change-expired-password;

<PRIV> – необходимый уровень привилегий, принимает значение [1..15];

<COMMAND> – поддерево команд, задается строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

Установить для поддерева команд «show» корневого командного режима необходимый уровень привилегий 2. Команды поддерева «show interfaces» оставить с уровнем привилегий 1.

```
rtt(config)# privilege root level 2 "show"
rtt(config)# privilege root level 1 "show interfaces"
```

radius-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
radius-server dscp <DSCP>
no radius-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

63

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# radius-server dscp 40
```

radius-server host

Данная команда используется для добавления RADIUS-сервера в список используемых серверов и перехода в командный режим RADIUS SERVER.

Использование отрицательной формы команды (no) удаляет заданный RADIUS-сервер.

Синтаксис

```
[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес RADIUS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# radius-server host 10.100.100.1
rtt(config-radius-server)#
```

radius-server host

Данная команда используется для добавления RADIUS-сервера в профиль RADIUS-серверов.

Использование отрицательной формы команды (no) удаляет заданный RADIUS-сервер из профиля.

Синтаксис

```
[no] radius-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес RADIUS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес RADIUS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER-PROFILE

Пример

```
rtt(config-aaa-radius-profile)# radius-server host 10.100.100.1
```

radius-server retransmit

Данной командой задаётся количество перезапросов к последнему активному RADIUS-серверу, которое будет выполнено перед выполнением запросов к следующим RADIUS-серверам в списке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
radius-server retransmit <COUNT>
no radius-server retransmit
```

Параметры

<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# radius-server retransmit 5
```

radius-server timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
radius-server timeout <SEC>
no radius-server timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# radius-server timeout 5
```

retransmit

Данной командой задаётся количество перезапросов к RADIUS-серверу, которое будет выполнено перед выполнением запросов к следующим RADIUS-серверам в списке.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
retransmit <COUNT>
```

```
no retransmit
```

Параметры

<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10].

Значение по умолчанию

Не задан, используется значение глобального параметра, описанного в разделе radius-server retransmit.

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config)# retransmit 5
```

security passwords default-expired

Данной командой включается запрос на смену пароля по умолчанию для пользователя admin.

Использование отрицательной формы команды (no) отключает запрос на смену пароля по умолчанию.

Синтаксис

```
[no] security passwords default-expired
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Запрос на смену пароля по умолчанию отключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords default-expired
```

security passwords history

Данной командой включается режим запрета на использование ранее установленных паролей локальных пользователей. В качестве параметра указывается количество паролей, сохраняемых в памяти маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords history <COUNT>
no security passwords history
```

Параметры

<COUNT> – количество паролей, сохраняемых в памяти маршрутизатора [0..15]. При уменьшении данного значения лишние, более старые пароли удаляются.

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords history 5
```

security passwords lifetime

Данной командой устанавливается время действия пароля локального пользователя. При попытке подключения пользователя с истекшим паролем пользователь будет направлен в режим принудительной смены пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords lifetime <TIME>
no security passwords lifetime
```

Параметры

<TIME> – интервал времени действия пароля в днях, принимает значения [1..365].

Значение по умолчанию

Время действия пароля локального пользователя не ограничено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords lifetime 30
```

security passwords lower-case

Данной командой устанавливается минимальное количество строчных букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords lower-case <COUNT>
no security passwords lower-case
```

Параметры

<COUNT> – минимальное количество строчных букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [0..32].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords lower-case 2
```

security passwords max-length

Данной командой устанавливается ограничение на максимальную длину пароля локального пользователя и ENABLE-пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords max-length <NUM>
no security passwords max-length
```

Параметры

<NUM> – максимальное количество символов в пароле, задается в диапазоне [1..32].

Значение по умолчанию

32

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords max-length 30
```

security passwords min-length

Данной командой устанавливается ограничение на минимальную длину пароля локального пользователя и ENABLE-пароля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords min-length <NUM>
no security passwords min-length
```

Параметры

<NUM> – минимальное количество символов в пароле, задается в диапазоне [1..32].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords min-length 10
```

security passwords numeric-count

Данной командой устанавливается минимальное количество цифр в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords numeric-count <COUNT>
no security passwords numeric-count
```

Параметры

<COUNT> – минимальное количество цифр в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [0..32].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords numeric-count 2
```

security passwords special-case

Данной командой устанавливается минимальное количество специальных символов в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords special-case <COUNT>
no security passwords special-case
```

Параметры

<COUNT> – минимальное количество специальных символов в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [0..32].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords special-case 2
```

security passwords symbol-types

Данной командой устанавливается минимальное количество типов символов, которые должны присутствовать в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords symbol-types <COUNT>
no security passwords symbol-types
```

Параметры

<COUNT> – минимальное количество типов символов в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [1..4].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords symbol-types 2
```

security passwords upper-case

Данной командой устанавливается минимальное количество прописных (заглавных) букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords upper-case <COUNT>
no security passwords upper-case
```

Параметры

<COUNT> – минимальное количество прописных (заглавных) букв в пароле локального пользователя, ENABLE-пароле, имени SMNPv3-пользователя и SMNPv1/SMNPv2 с комьюнити [0..32].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security passwords upper-case 2
```

security snmp-community max-length

Данной командой устанавливается ограничение на максимальную длину SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security snmp-community max-length <NUM>
no security snmp-community max-length
```

Параметры

<NUM> – максимальное количество символов в комьюнити, задается в диапазоне [1..128].

Значение по умолчанию

128

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security snmp-community max-length 30
```

security snmp-community min-length

Данной командой устанавливается ограничение на минимальную длину SMNPv1/SMNPv2 с комьюнити.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security passwords min-length <NUM>
no security passwords min-length
```

Параметры

<NUM> – минимальное количество символов в комьюнити, задается в диапазоне [1..128].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security snmp-community min-length 10
```

show aaa accounting

Данная команда позволяет просмотреть настроенные параметры учета.

Синтаксис

```
show aaa accounting
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show aaa accounting
Login :          radius
Commands :       tacacs
```

show aaa authentication

Данная команда позволяет просмотреть списки способов аутентификации пользователей, а также активные списки каждого типа терминалов.

Синтаксис

```
show aaa authentication
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show aaa authentication
   Login Authentication Method Lists
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
List               Methods
----------------   --------------------------------
default            local
   Enable Authentication Method Lists
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
List               Methods
----------------   --------------------------------
default            enable
   Lines configuration
   ~~~~~~~~~~~~~~~~~~~
Line        Login method list                Enable method list
---------   ------------------------------   --------------------------------
console     default                          default
telnet      default                          default
ssh         default                          default
```

show aaa ldap-servers

Данная команда позволяет просмотреть параметры LDAP-серверов.

Синтаксис

```
show aaa ldap-servers
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show aaa ldap-servers
Base DN:                        dc=example,dc=com
Root DN:                        cn=admin,dc=example,dc=com
Root password:                  CDE65039E5591FA3
Naming attribute:               uid
Privilege level attribute:      priv-lvl
User object class:              posixAccount
DSCP:                           63
Bind timeout:                   3
```

```
Search timeout:               0
Search scope:                 subtree
IP Address                         Port           Priority
-------------------------------   ------------   ------------
10.100.100.1                       389            1
```

show aaa radius-servers

Данная команда позволяет просмотреть параметры RADIUS-серверов.

Синтаксис

```
show aaa radius-servers
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show aaa radius-servers
Timeout:     3
Retransmit:  1
DSCP:        63
IP Addres        Timeout      Priority     Usage        Key
------------    ----------   ----------   ----------   --------------------------
2.2.2.2             --           1            all          9DA7076CA30B5FFE0DC9C4
2.4.4.4             --           1            all          9DA7076BA30B4EFCE5
```

show aaa tacacs-servers

Данная команда позволяет просмотреть параметры TACACS-серверов.

Синтаксис

```
show aaa tacacs-servers
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show aaa tacacs-servers
Timeout :       3
DSCP:          63
IP Address               Port           Priority       Key
-----------------------  -------------  -------------  --------------------------------
10.100.100.1             49             1              CDE65039E5591FA3
10.100.100.5             49             10             CDE65039E5591FA3
```

show users

Данная команда позволяет просмотреть активные сессии пользователей системы.

Синтаксис

```
show users
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show users
 SID    User name             Logged in at         Protocol           Host                Timers Login/Priv   Level
 ----   --------------------  -----------------    -----------------  ------------------  -----------------   -----
 0 *    admin                 2023-06-26 15:47:38  SSH                192.168.1.44        00:29:59/00:00:00   15
 1      admin                 2023-06-26 15:47:38  Telnet             192.168.1.44        00:25:08/00:00:00   15
 2      admin                 2023-06-26 15:47:38  Console            Console             00:29:56/00:00:00   15
```

show users accounts

Данная команда позволяет просмотреть конфигурацию пользователей системы.

Синтаксис

```
show users accounts
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show user accounts
Name                              Password                              Privilege
--------------------------------  ------------------------------------  ---------
admin                             $6$1sxrvGaV8Za8oX/K$YNel5xYPZ4cj      15
                                  bemYWYNpQBQKDxWE9v0aoKgQ
                                  kRCEb0EMNuusO9Kmg7UBs7nA3buEM87e
                                  Eu.rA6tZq0
techsupport                       $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV      15
                                  9jHcp. 9lweQaSldw7ZtUr
                                  uH66uZx9.EBASff//hUj8ObUaC484TNR
                                  x.
remote                            $6$YfwntIwU$ah7UxPZTemKhjpSWvVsV      1
                                  9jHcp.kqFAK.vmvyY9lweQaSldw7ZtUr
                                  uH66uZx9.EBASff//hUj8ObUaC484TNR
                                  x.
operator                          $6$eILpbbyRxedCzvVD$4RHP08mjXvNf      1
                                  urX7V/ULCZ1oHIWMwE6h5f
                                  zgwZQUZcPoZCEyaqQQqCicRMRuPwhxrQ
                                  bvGChWreW1
```

show users blocked

Данная команда позволяет просматривать список пользователей, для которых был введен неправильный пароль. Пользователь удаляется из списка после ввода правильного пароля при аутентификации.

Синтаксис

```
show users blocked [ <NAME> ]
```

Параметры

<NAME> – имя пользователя, для которого необходимо отобразить статистику неправильных попыток аутентификации, задаётся строкой до 31 символа.

Без указания имени пользователя отображается вся таблица неправильных попыток аутентификации.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show users blocked
User name              Failures   Latest failure        From
--------------------   --------   -------------------   ----------------
tester                 4          2017-09-10 08:29:42   0.0.0.0
```

source-address

Данной командой определяется IPv4/IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых пакетах на конфигурируемый AAA-сервер.

Использование отрицательной формы команды (no) удаляет указанный IPv4/IPv6-адрес источника.

Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no source-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

Пример

```
rtt(config-radius-server)# source-address 220::71
```

source-interface

Данной командой определяется интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться в качестве IPv4/IPv6-адреса источника в отправляемых пакетах на конфигурируемый AAA-сервер.

Использование отрицательной формы команды (no) удаляет указанный интерфейс или туннель.

Синтаксис

```
source-interface { <IF> | <TUN> }
no source-interface
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

CONFIG-TACACS-SERVER

CONFIG-LDAP-SERVER

Пример

```
rtt(config-radius-server)# source-interface gigabitethernet 1/0/1
```

system configuration-exclusively

Данной командой включается ограничение количества сессий CLI до одной.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
[no] system configuration-exclusively
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# system configuration-exclusively
```

tacacs-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов TACACS-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
tacacs-server dscp <DSCP>
no tacacs-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

63

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# tacacs-server dscp 40
```

tacacs-server host

Данная команда используется для добавления TACACS-сервера в список используемых серверов и перехода в командный режим TACACS SERVER.

Использование отрицательной формы команды (no) удаляет заданный TACACS-сервер.

Синтаксис

```
[no] tacacs-server host { <ADDR> | <IPV6-ADDR> } [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<ADDR> – IP-адрес TACACS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IPV6-ADDR> – IPv6-адрес TACACS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# tacacs-server host 10.100.100.1
rtt(config-tacacs-server)#
```

tacacs-server timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что TACACS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
tacacs-server timeout <SEC>
no tacacs-server timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3 секунды.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# tacacs-server timeout 5
```

timeout

Данной командой задаётся интервал, по истечении которого устройство считает, что RADIUS-сервер недоступен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timeout <SEC>
no timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

Не задан, используется значение глобального таймера, описанного в разделе radius-server timeout.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config-radius-server)# timeout 7
```

usage

Данная команда определяет тип соединений для аутентификации которых будет использоваться RADIUS-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
usage { all | aaa | auth | acct | pptp | l2tp }
no usage
```

Параметры

all – все типы соединений;

aaa – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета telnet-, ssh-, console-сессий;

auth – RADIUS-сервер будет использоваться для аутентификации и авторизации telnet-, ssh-, console-сессий;

acct – RADIUS-сервер будет использоваться для учета telnet-, ssh-, console-сессий;

pptp – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета удаленных пользователей, подключающихся по протоколу PPTP;

l2tp – RADIUS-сервер будет использоваться для аутентификации, авторизации и учета удаленных пользователей, подключающихся по протоколу L2TP over IPsec.

Значение по умолчанию

all

Необходимый уровень привилегий

15

Командный режим

CONFIG-RADIUS-SERVER

Пример

```
rtt(config-radius-server)# usage pptp
```

username

Данной командой выполняется добавление пользователя в локальную базу пользователей и осуществляется переход в режим настройки параметров пользователя.

Использование отрицательной формы команды (no) удаляет пользователя из системы.

Синтаксис

```
[no] username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пользователи.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# username test
rtt(config-user)#
```

12. КОНФИГУРИРОВАНИЕ И МОНИТОРИНГ ИНТЕРФЕЙСОВ

Порядок именования интерфейсов маршрутизатора описан в разделе Типы и порядок именования интерфейсов маршрутизатора.

Команды, введенные в режиме конфигурирования интерфейса (группы интерфейсов), применяются к выбранному интерфейсу (группе интерфейсов).

12.1. Общие команды

clear interfaces counters

Данной командой осуществляется сброс счетчиков заданного системного интерфейса или группы интерфейсов.

Синтаксис

```
clear interfaces counters [<IF>]
```

Параметры

<IF> – наименование системного интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Можно указать несколько интерфейсов перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы интерфейсов, то будут очищены счетчики всех интерфейсов заданной группы. При выполнении команды без параметра будут очищены счетчики всех системных интерфейсов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear interfaces counters gigabitethernet 1/0/5
```

description

Данная команда используется для изменения описания конфигурируемого интерфейса.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-SERVICE-PORT

CONFIG-CELLULAR-PROFILE

CONFIG-CELLULAR-MODEM

CONFIG-VOICE-PORT

CONFIG-VOIP-PROFILE

Пример

```
rtt(config-if-gi)# description "Uplink interface"
```

history statistics

Данной командой включается запись статистики использования текущего интерфейса.

Использование отрицательной формы команды (no) отключает запись статистики использования текущего интерфейса.

Синтаксис

```
[no] history statistics
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-IF-MULTILINK

CONFIG-IF-E1

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-if-gi)# history statistics
rtt(config-if-gi)#
```

interface

Данная команда позволяет перейти в режим конфигурирования одного или более интерфейсов.

Использование отрицательной формы команды (no) восстанавливает настройки интерфейса по умолчанию.

Синтаксис

```
[no] interface <IF>
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

На маршрутизаторах R800 не поддерживается работа модулей ToPGATE-WAN-E1 с аппаратной версией (hardware revision) 812.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Переход в режим конфигурирования Ethernet-интерфейса gi 1/0/20:

```
rtt(config)# interface gigabitethernet 1/0/20
rtt(config-if-gi)#
```

Пример 2

Переход в режим конфигурирования Ethernet-интерфейса te 1/0/2:

```
rtt(config)# interface tengigabitethernet 1/0/2
rtt(config-if-te)#
```

Пример 3

Переход в режим конфигурирования виртуального интерфейса:

```
rtt(config)# interface loopback 5
rtt(config-loopback)#
```

Пример 4

Переход в режим конфигурирования саб-интерфейса:

```
rtt(config)# interface gigabitethernet 1/0/20.20
rtt(CONFIG-IF-SUB)#
```

Пример 5

Переход в режим конфигурирования интерфейса port-channel 2:

```
rtt(config)# interface port-channel 2
rtt(CONFIG-IF-PORT-CHANNEL)#
```

Пример 6

Переход в режим конфигурирования интерфейса e1 1/0/1:

```
rtt(config)# interface e1 1/0/1
rtt(config-if-e1)#
```

Пример 7

Переход в режим конфигурирования интерфейса multilink 1:

```
rtt(config)# interface multilink 1
rtt(CONFIG-IF-MULTILINK)#
```

ip tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum Segment Size) как для транзитных, так и для локально созданных TCP-сессий. Значение MSS для сессии меняется только при условии, что оно выше, чем настроено на интерфейсе.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

Синтаксис

```
ip tcp adjust-mss <MSS>
no ip tcp adjust-mss
```

Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

Значение по умолчанию

1460

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# ip tcp adjust-mss 1400
```

load-average

Данной командой устанавливается интервал времени, по которому происходит расчет утилизации интерфейса.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
load-average <TIME>
no load-average
```

Параметры

<TIME> – интервал в секундах, принимает значения [5..150].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-if-gi)# load-average 30
```

mode

Данной командой устанавливается режим работы физического/агрегированного интерфейса.

Использование отрицательной формы команды (no) устанавливает значение режима по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы физического/агрегированного интерфейса, принимает значения:

- switchport – устанавливает L2-режим. Возможно разрешение vlan, но запрещает назначение IP-адреса и создание саб/qinq-интерфейсов.
- routerport – устанавливает L3-режим. Возможно назначение IP-адреса и создание саб/qinq-интерфейсов, но запрещено разрешение vlan на интерфейсе.
- hybrid – возможно разрешение vlan, назначение IP-адреса и создание саб/qinq-интерфейсов.

Значение «hybrid» доступно только на маршрутизаторах R800.

Значение по умолчанию

routerport

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# mode switchport
```

mtu

Данной командой указывается размер MTU (Maximum Transmition Unit) для интерфейсов.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

```
mtu <MTU>
no mtu
```

Параметры

<MTU> – значение MTU в байтах, принимает значения в диапазоне: [552..10000].

Для Serial, E1 и multilink-интерфейсов – [552-1500] для всех моделей маршрутизаторов.

Значения MTU более 1500 можно выставлять только при включенной поддержке Jumbo-фреймов, описанной в разделе system jumbo-frames.

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-if-gi)# mtu 1400
```

rate-limit arp-broadcast

Данной командой включается ограничение приема arp-запросов на bridge-интерфейсе.

При использовании отрицательной формы команды (no) отключается ограничение приема arp-запросов на bridge-интерфейсе.

Синтаксис

```
[no] rate-limit arp-broadcast
```

Параметры

Отсутствуют.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# rate-limit arp-broadcast
```

rate-limit arp-broadcast pps

Данной командой устанавливается максимальный порог приема arp-запросов на bridge-интерфейсе. Данный параметр работает только при включенном режиме ограничения приема arp-запросов на bridge-интерфейсе.

При использовании отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
rate-limit arp-broadcast pps <PPS>
no rate-limit arp-broadcast pps
```

Параметры

<PPS> – количество arp-запросов, задается в диапазоне [1..65535].

Значение по умолчанию

100

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# rate-limit arp-broadcast pps 2000
```

show interfaces counters

Командой выполняется просмотр счетчиков на системных интерфейсах: портах, саб-интерфейсах, группах агрегации, сетевых мостах.

Синтаксис

```
show interfaces counters [ <IF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения счетчиков интерфейсов в формате вывода без включенного кластера.

Команда отображает счётчики для портов маршрутизатора, саб-интерфейсов и туннельных интерфейсов.

Можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены счетчики всех интерфейсов заданной группы. Если задан определённый интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны счетчики всех системных интерфейсов.

При включенном кластере отображает счетчики для все юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces counters gigabitethernet 1/0/4-6
Interface       UC recv         Bytes recv      Errors recv     MC recv
-------------   -------------   -------------   -------------   -------------
gi1/0/4         0               0               0               0
gi1/0/5         0               0               0               0
gi1/0/6         0               0               0               0
Interface       UC sent         Bytes sent      Errors sent
-------------   -------------   -------------   -------------
gi1/0/4         0               0               0
gi1/0/5         1138            393748          0
gi1/0/6         0               0               0
rtt# show interfaces counters gigabitethernet 1/0/4
 Packets received:               0
 Bytes received:                 0
 Dropped on receive:             0
 Receive errors:                 0
 Multicasts received:            0
 Receive length errors:          0
 Receive buffer overflow errors: 0
 Receive CRC errors:             0
 Receive frame errors:           0
 Receive FIFO errors:            0
 Receive missed errors:          0
 Receive compressed:             0
 Packets transmitted:            0
 Bytes transmitted:              0
 Dropped on transmit:            0
 Transmit errors:                0
 Transmit aborted errors:        0
 Transmit carrier errors:        0
 Transmit FIFO errors:           0
 Transmit heartbeat errors:      0
 Transmit window errors:         0
 Transmit comressed:             0
 Collisions:                     0
```

show interfaces description

Команда используется для просмотра описания системных интерфейсов.

Синтаксис

```
show interfaces description [ <IF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения описания интерфейсов в формате вывода без включенного кластера.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены описания всех интерфейсов заданной группы. При выполнении команды без параметра будут показаны описания всех системных интерфейсов.

При включенном кластере отображает описание системных интерфейсов для все юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces description gigabitethernet 1/0/4-5
Interface       Admin   Link    Description
                State   State
-------------   -----   -----   ----------------------------------
gi1/0/4         Up      Down    Link to NSK
gi1/0/5         Up      Down    Link to MSK
```

show interfaces history

Команда используется для просмотра статистики использования интерфейса.

Синтаксис

```
show interfaces history [<IF>] [timer <TIMER>]
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TIMER > – не обязательный ключ timer. В качестве параметров для данного ключа могут выступать:

- hours отображает историю за последние 72 часа,

- minutes отображает историю за последние 60 минут,
- seconds отображает историю за последние 60 секунд,
- при отсутствии ключа timer выводятся 3 таблицы истории использования туннеля(ей).

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# show interfaces history gi 1/0/1 timer minutes
gi1/0/1
Last 60 minutes:
Timer  Recv utilization, Kbit/s  Sent utilization, Kbit/s  Recv errors  Sent errors  Output
drops
------ ------------------------ ------------------------ ------------ ------------ -------
-------
0-1    240            16             0       0       0
1-2    961            64             0       0       0
2-3    962            64             0       0       0
3-4    962            64             0       0       0
4-5    960            64             0       0       0
5-6    961            64             0       0       0
6-7    719            64             0       0       0
7-8    960            64             0       0       0
8-9    800            65             0       0       0
9-10   962            64             0       0       0
10-11  865            64             0       0       0
11-12  962            64             0       0       0
12-13  817            65             0       0       0
13-14  962            65             0       0       0
14-15  961            65             0       0       0
15-16  880            60             0       0       0
16-17  960            63             0       0       0
17-18  0              0              0       0       0
18-19  0              0              0       0       0
19-20  0              0              0       0       0
20-21  0              0              0       0       0
21-22  0              0              0       0       0
```

show interfaces status

Команда используется для просмотра состояния системных интерфейсов.

Синтаксис

```
show interfaces status [<IF>] [ unit { <RANGE-ID> | local } ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения состояния системных интерфейсов в формате вывода без включенного кластера.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.

При включенном кластере отображает состояние системных интерфейсов для все юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces status gigabitethernet 1/0/1-2
Interface            Admin   Link    MTU     MAC address         Last change    Mode
                     State   State                                (d,h:m:s)
-------------------  -----   -----   -----   -----------------   ------------   -----
gi1/0/1               Up     Up      1500    a8:f9:4b:af:20:f6   25,06:52:50 routerport
gi1/0/2               Up     Up      1500    a8:f9:4b:af:20:f7   25,06:53:03 routerport
```

show interfaces utilization

Команда используется для просмотра текущей нагрузки на физических интерфейсах.

Синтаксис

```
show interfaces utilization [ <IF> ]
```

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будет отображена текущая нагрузка для всех интерфейсов заданной группы. При выполнении команды без параметра будет показана текущая нагрузка для всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces utilization gigabitethernet 1/0/3-5,1/0/9
Port       Period, s     Sent,         Recv,         Frames Sent   Frames Recv
                         Kbit/s        Kbit/s
--------   -----------   -----------   -----------   -----------   -----------
gi1/0/3    5             0             0             0             0
gi1/0/4    5             0             0             0             0
gi1/0/5    5             0             0             0             0
gi1/0/9    5             0             0             0             0
```

show system jumbo-frames

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R200 и R800.

Команда используется для просмотра текущего состояния и состояния после перезагрузки устройства функции jumbo-фреймов.

Синтаксис

```
show system jumbo-frames
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show system jumbo-frames
Jumbo frames are disabled
Jumbo frames will be disabled after reset
```

shutdown

Данной командой отключается конфигурируемый интерфейс.

Использование отрицательной формы команды (no) включает конфигурируемый интерфейс.

Синтаксис

```
[no] shutdown
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-gi)# shutdown
```

Конфигурируемый интерфейс отключен.

speed

Данной командой устанавливается значение скорости для конфигурируемого интерфейса, группы интерфейсов. Командой могут быть установлены следующие режимы: 10 Мбит/с, 100 Мбит/с, 1000 Мбит/с, 10 Гбит/с, 25 Гбит/с, 40 Гбит/с или auto.

Использование отрицательной (no) формы команды устанавливает значение по умолчанию.

Синтаксис

```
speed <SPEED> <DUPLEX>
no speed
```

Параметры

<SPEED> – значение скорости:

- 10M – значение скорости 10 Мбит/с;
- 100M – значение скорости 100 Мбит/с;
- 1000M – значение скорости 1000 Мбит/с;
- 10G – значение скорости 10 Гбит/с;
- auto – автоматический выбор режима (недоступно для 10G-интерфейсов).

<DUPLEX> – режим работы приемопередатчика, принимает значения:

- full-duplex – дуплекс;
- half-duplex – полудуплекс.

Значение по умолчанию

auto

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-PORT-CHANNEL

Пример 1

```
rtt(config-if-te)# speed 10G
```

Установлен скоростной режим интерфейса 10 Гбит/с.

Пример 2

```
rtt(config-if-gi)# speed 10M full-duplex
```

Установлен скоростной режим интерфейса 10 Мбит/с, дуплекс.

switchport dot1q ethertype egress stag

Данной командой настраивается EtherType для сервисного VLAN в исходящих пакетах.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] switchport dot1q ethertype egress stag { 802.1q | 802.1ad }
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

802.1q (0x8100)

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport dot1q ethertype egress stag 802.1ad
```

switchport mode

Данная команда используется для задания режима работы интерфейса с VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport mode <MODE>
no switchport mode
```

Параметры

<MODE> – режим работы:

- access – интерфейс доступа, нетегированный интерфейс для одной VLAN;
- trunk – интерфейс, принимающий только тегированный трафик за исключением одного VLAN, который может быть добавлен с помощью команды *switchport trunk native vlan*, описанной в switchport trunk native-vlan;
- general – интерфейс переключается в режим general;
- e1 – физический интерфейс переключается в режим E1 (не применима для агрегированных интерфейсов).

Режим работы «Access» доступен только на маршрутизаторах R100/200. Режим работы «Trunk» доступен только на маршрутизаторах R800.

Значение по умолчанию

access

general

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport mode trunk
```

system jumbo-frames

Данной командой включается поддержка Jumbo-фреймов. Для вступления изменений в силу требуется перезагрузка устройства.

Использование отрицательной формы команды (no) выключает поддержку Jumbo-фреймов.

Синтаксис

```
[no] system jumbo-frames
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system jumbo-frames
```

12.2. Физические интерфейсы

snmp trap link-status

Данная команда используется для включения отправки snmp-trap о включении/выключении туннеля.

Использование отрицательной формы команды (no) отключает отправку snmp-trap о включении/отключении туннеля.

Синтаксис

```
[no] snmp trap link-status
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Включено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

Пример

```
rtt(config-if-gi)# snmp trap link-status
```

show interfaces protected-port

Команда используется для просмотра физических интерфейсов в режиме изоляции по группам.

Синтаксис

```
show interfaces protected-ports [ <IF> ]
```

Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces protected-ports
Interface   State           Community
---------   -------------   ---------
gi1/0/5     Protected       4
```

show interfaces sfp

Команда используется для просмотра информации об SFP-трансиверах.

Синтаксис

```
show interfaces sfp [ <IF> ]
```

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора. В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces sfp
Interface 'te1/0/1':
 SFP present:       Yes
 Connector Type:    LC
 Type:              SFP/SFP+
 Compliance code:   10G BASE-SR
 Laser wavelength:  850 nm
 Transfer distance: 300.00 m
 Vendor OUI:        24:00:00
 Vendor name:       Modultech
 Vendor PN:         MT-PP-85192-SR
 Vendor SN:         M1204011007
```

```
 Vendor date:        04.05.12
 Vendor revision:    1.0
 DDM supported:      Yes
 Temperature:        40.562 C
 Voltage:            3.3364 V
 Current:            6.004 mA
 RX Power:           0.0001 mW / -40.0000 dBm
 TX Power:           0.4398 mW / -3.5674 dBm
 RX LOS:             Yes
 TX Fault:           No
 TX Disable:         No
 Soft TX Disable:    No
Interface 'te1/0/2':
 SFP present:        Yes
 Connector Type:     SC
 Type:               SFP/SFP+
 Compliance code:    1000BASE-LX
 Laser wavelength:   1310 nm
 Transfer distance:  20.00 km
 Vendor OUI:         --
 Vendor name:        OEM
 Vendor PN:          APSB35123CXS20
 Vendor SN:          SG35224701333
 Vendor date:        12.12.12
 Vendor revision:    1.00
 DDM supported:      No
```

show interfaces switch-port configuration

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Командой выполняется просмотр параметров конфигурации физических интерфейсов.

Синтаксис

```
show interfaces switch-port configuration [ <IF> ]
```

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Можно указать несколько интерфейсов перечислением через запятую либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будут отображены параметры всех интерфейсов заданной группы. При выполнении команды без параметра будут показаны параметры всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces switch-port configuration gigabitethernet 1/0/5-7
Port       Media     Duplex   Speed      Neg        Flow      Admin   Back
                                                    control   State   Pressure
--------   -------   ------   --------   --------   -------   -----   --------
gi1/0/5    none      Half     10 Mbps    Enabled    Off       Up      Disabled
gi1/0/6    none      Half     10 Mbps    Enabled    Off       Up      Disabled
gi1/0/7    none      Half     10 Mbps    Enabled    Off       Up      Disabled
```

show interfaces switch-port status

Команда используется для просмотра состояния физических интерфейсов.

Синтаксис

```
show interfaces switch-port status [ <IF> ]
```

Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Можно указать несколько интерфейсов перечислением через запятую «,» либо указать диапазон интерфейсов через дефис «-». Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces switch-port status
Port       Media     Duplex   Speed   Neg        Flow   Link    Back       MDI     Port
                                                  ctrl   State   Pressure   Mode    Mode
--------   ------    ------   -----   --------   ----   -----   --------   -----   --
gi1/0/1    --        --       --      Enabled    --     Down    --         --      access
gi1/0/2    --        --       --      Enabled    --     Down    --         --      access
gi1/0/3    --        --       --      Enabled    --     Down    --         --      access
gi1/0/4    --        --       --      Enabled    --     Down    --         --      access
gi1/0/5    --        --       --      Enabled    --     Down    --         --      access
gi1/0/6    --        --       --      Enabled    --     Down    --         --      access

rtt# show interfaces switch-port status gigabitethernet 1/0/2
Interface       gigabitethernet 1/0/2
  Status:       up
```

```
  Media:         copper
  Speed:         100 Mbps
  Duplex:        full
  Flow control: no
  MAC address:  a8:f9:b5:00:00:25
MAC status:
  Buffers full:            no
  Doing back pressure:     no
  Sending PAUSE frames:    no
  Receiving PAUSE frames:  no
  Auto-Negotiation done:   yes
  Sync fail:               no
```

switchport community

Данной командой интерфейс добавляется в группу изоляции. Данная команда актуальна, только если порт находится в режиме изоляции по группам.

Использование отрицательной формы команды (no) удаляет интерфейс из группы изоляции.

Синтаксис

```
switchport community <ID>
no switchport community
```

Параметры

<ID> – идентификатор группы, принимает значения в диапазоне [1..30].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport community 10
```

switchport protected

Данной командой на интерфейсе включается функция Private VLAN и указывается интерфейс, на который могут отправляться принятые пакеты.

Использование отрицательной формы команды (no) отключает функцию Private VLAN.

Синтаксис

```
switchport protected <IF>
no switchport protected
```

Параметры

<IF> – наименование интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Значение по умолчанию

Функция Private VLAN отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport protected gigabitethernet 1/0/1
```

switchport protected-port

Данной командой интерфейс переводится в режим изоляции по группам. В данном режиме обмен трафиком между интерфейсами одной группы разрешен, обмен трафиком между интерфейсами разных групп запрещен, обмен трафиком между изолированными и неизолированными интерфейсами разрешен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] switchport protected-port
```

Параметры

Отсутствуют.

Значение по умолчанию

Интерфейс не изолирован.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt:rtt(config-if-gi)# switchport protected-port
```

threshold rx-utilization

Данная команда используется для задания порога входящей нагрузки интерфейса для отправки snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

Синтаксис

```
threshold rx-utilization <TH-HIGH> clear <TH-LOW>
no threshold rx-utilization
```

Параметры

<TH-HIGH> – порог в процентах для отправки snmp-trap eltexInterfaceRxUtilizationHigh;

<TH-LOW> – порог в процентах для отправки snmp-trap eltexInterfaceRxUtilizationHighOk.

Значение по умолчанию

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# threshold rx-utilization 90 clear 80
```

threshold rx-utilization monitoring

Данная команда используется для включения отправки snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

Использование отрицательной формы команды (no) отключает отправку snmp-trap eltexInterfaceRxUtilizationHigh и eltexInterfaceRxUtilizationHighOk.

Синтаксис

```
[no] threshold rx-utilization monitoring
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# threshold rx-utilization monitoring
```

threshold tx-utilization

Данная команда используется для задания порогов исходящей нагрузки интерфейса для отправки snmp-trap eltexInterface T xUtilizationHigh и eltexInterface T xUtilizationHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

Синтаксис

```
threshold tx-utilization <TH-HIGH> clear <TH-LOW>
no threshold tx-utilization
```

Параметры

<TH-HIGH> – порог в процентах для отправки snmp-trap eltexInterfaceTxUtilizationHigh;

<TH-LOW> – порог в процентах для отправки snmp-trap eltexInterfaceTxUtilizationHighOk.

Значение по умолчанию

<TH-HIGH> – 90%;

<TH-LOW> – 85%.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# threshold tx-utilization 90 clear 80
```

threshold tx-utilization monitoring

Данная команда используется для включения отправки snmp-trap eltexInterface T xUtilizationHigh и eltexInterface T xUtilizationHighOk.

Использование отрицательной формы команды (no) отключает отправку snmp-trap eltexInterface T xUtilizationHigh и eltexInterface T xUtilizationHighOk.

Синтаксис

```
[no] threshold tx-utilization monitoring
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# threshold tx-utilization monitoring
```

12.3. Агрегированные интерфейсы

channel-group

Данной командой физический интерфейс включается в группу агрегации каналов.

Использование отрицательной формы команды (no) удаляет интерфейс из группы агрегации каналов.

Синтаксис

```
channel-group <ID> mode <MODE>
no channel-group
```

Параметры

<ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].

<MODE> – режим формирования группы агрегации каналов:

- auto – добавить интерфейс в динамическую группу агрегации с поддержкой протокола LACP;
- on – добавить интерфейс в статическую группу агрегации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# channel-group 6 mode auto
```

lacp port-priority

Данной командой устанавливается LACP-приоритет интерфейса Ethernet.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
lacp port-priority <PRIORITY>
no lacp port-priority
```

Параметры

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# lacp port-priority 5000
```

lacp system-priority

Данной командой устанавливается приоритет системы для протокола LACP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
lacp system-priority <PRIORITY>
no lacp system-priority
```

Параметры

<PRIORITY> – приоритет, указывается в диапазоне [1..65535].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# lacp system-priority 5000
```

lacp timeout

Данной командой устанавливается административный таймаут протокола LACP.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
lacp timeout { short | long }
no lacp timeout
```

Параметры

long – длительное время таймаута (90 секунд);

short – короткое время таймаута (3 секунды).

Значение по умолчанию

long

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# lacp timeout short
```

port-channel load-balance

Данной командой устанавливается механизм балансировки нагрузки для групп агрегации каналов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port-channel load-balance {src-dst-mac-ip|src-dst-mac|src-dst-ip|src-dst-mac-ip-port}
no port-channel load-balance
```

Параметры для R800

src-dst-mac – механизм балансировки основывается на MAC-адресе отправителя и получателя;

src-dst-ip – механизм балансировки основывается на IP-адресе отправителя и получателя;

src-dst-mac-ip – механизм балансировки основывается на MAC-адресе и IP-адресе отправителя и получателя;

src-dst-ip-port – механизм балансировки основывается на IP-адресах и tcp/udp-портах отправителя и получателя;

src-dst-mac-ip-port – механизм балансировки основывается на MAC-адресе, IP-адресе и порте отправителя и получателя.

Параметры для R100

active-backup – весь трафик отправляется в один интерфейс и перенаправляется в другой, если первый перешел в состояние down;

src-dst-mac – механизм балансировки основывается на MAC-адресах отправителя и получателя;

src-dst-mac-ip – механизм балансировки основывается на MAC-адресах и IP-адресах отправителя и получателя;

src-dst-ip-port – механизм балансировки основывается на IP-адресах и tcp/udp-портах отправителя и получателя.

Значение по умолчанию

src-dst-mac

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# port-channel load-balance src-dst-mac-ip
```

show interfaces port-channel

Данная команда используется для просмотра информации о членах группы агрегации каналов.

Синтаксис

```
show interfaces port-channel { <ID> | <UNIT>/<ID> }
```

Параметры

<ID> – порядковый номер группы агрегации каналов, принимает значения в диапазоне [1..12].
<UNIT> – номер текущего юнита устройства, принимает значения в диапазоне [1..4].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show interfaces port-channel 1
load-balance: src-dst-mac
Channels           Ports
----------------   ------------------------------------------------
port-channel 1     gi1/0/1
```

Пример 2

```
rtt# show interfaces port-channel 1/1
```

```
Channels           Ports
---------------    ---------------------------------------------------------
port-channel 1/1   gi1/0/1
```

show lacp counters

В текущей версии ПО данный функционал поддерживается только на маршрутизаторах R800.

Данная команда используется для просмотра статистики работы LACP-протокола для интерфейса Ethernet.

Синтаксис

```
show lacp counters [ <IF> ]
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена статистика всех интерфейсов заданной группы. Если задан определенный интерфейс, то будет отображена детальная информация по данному интерфейсу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show lacp counters port-channel 1
Interface              Sent           Recv           Link failure
--------------------   ------------   ------------   ------------
po1                    0              2626           0
```

Пример 2

```
rtt# show lacp counters port-channel 1/1
Interface              Sent           Recv           Link failure
--------------------   ------------   ------------   ------------
po1/1                  0              2626           0
```

show lacp interfaces

Данная команда используется для просмотра информации о протоколе LACP.

Синтаксис

```
show lacp interfaces [ <IF> ]
```

Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена информация о LACP-протоколе для всех интерфейсов заданной группы.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show lacp interfaces port-channel 1
port-channel 1 [aggregator 1, active] ports count: 1
                  Actor Port             Partner Port
----------------  --------------------   --------------------
System Priority   32768                  1
System MAC        a8:f9:4b:aa:12:40      a8:f9:4b:83:01:80
Key               8000                   1
port-channel 1 [aggregator 2, backup] ports count: 1
                  Actor Port             Partner Port
----------------  --------------------   --------------------
System Priority   32768                  65535
System MAC        a8:f9:4b:aa:12:40      00:00:00:00:00:00
Key               8000                   FFFF
```

Пример 2

```
rtt# show lacp interfaces port-channel 1/1
port-channel 1/1 [aggregator 1, backup] ports count: 1
                  Actor Port             Partner Port
----------------  --------------------   --------------------
System Priority   32768                  1
System MAC        a8:f9:4b:af:a6:c0      a8:f9:4b:aa:12:40
Key               8000                   1

port-channel 1/1 [aggregator 2, backup] ports count: 1
                  Actor Port             Partner Port
----------------  --------------------   --------------------
System Priority   32768                  65535
System MAC        a8:f9:4b:af:a6:c0      a8:f9:4b:aa:12:40
Key               8000                   1
```

show lacp parameters

Данная команда используется для просмотра параметров настройки протокола LACP для интерфейса Ethernet.

Синтаксис

```
show lacp parameters [ <IF> ]
```

Параметры

<IF> – имя физического интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора. Возможно использование только физических интерфейсов.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены параметры всех интерфейсов заданной группы. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show lacp parameters gigabitethernet 1/0/1
Interface              Port Priority   Timeout    Link Mode
--------------------   -------------   --------   ----------
gi1/0/1                32768           Long       Active
```

12.4. E1/multilink

enable

Данной командой включается PPP-пользователь.

Использование отрицательной формы команды (no) отключает PPP-пользователя.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

PPP-пользователь отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPP-USER

Пример

```
rtt(config-ppp-user)# enable
```

ip tcp header-compression

Данной командой включается протокол сжатия tcp-заголовком. Протокол используется для улучшения производительности низкоскоростных каналов связи.

Использование отрицательной формы команды (no) отключает протокол сжатия tcp-заголовков.

Синтаксис

```
[no] ip tcp header-compression
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
(config-if-e1)# ip tcp header-compression
```

ip tcp compression-connections

Данной командой указывается количество одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для количества одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков.

Синтаксис

```
ip tcp compression-connections <NUMBER>
no ip tcp compression-connections
```

Параметры

<NUMBER> – количество одновременных tcp-соединений, для которых будет использоваться протокол компрессии tcp-заголовков. Может принимать значения [2..16].

Значение по умолчанию

16

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt:rtt(config-if-e1)# ip tcp compression-connections 32
rtt:rtt(config-if-e1)#
```

mrru

Данная команда определяет максимальный размер принимаемого пакета для MLPPP-интерфейса.

Использование отрицательной формы команды (no) устанавливает значение mrru по умолчанию.

Синтаксис

```
mrru { <MRRU> }
no mrru
```

Параметры

<MRRU> – максимальный размер принимаемого пакета для MLPPP-интерфейса, принимает значения в диапазоне [1500..10000].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-MULTILINK

Пример

```
rtt(CONFIG-IF-MULTILINK)# mrru 1700
```

password

Команда для установки пароля в открытой или зашифрованной форме определенному пользователю для аутентификации удаленной стороны. Пароль пользователя хранится в конфигурации в зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

Синтаксис

```
password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1 .. 64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPP-USER

Пример

```
rtt(config-ppp-user)# password ascii-text 01234567
```

ppp authentication chap

Данной командой включается CHAP-аутентификация.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
[no] ppp authentication chap
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp authentication chap
```

ppp chap hostname

Данной командой указывается имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp chap hostname <NAME>
no ppp chap hostname
```

Параметры

<NAME> – имя маршрутизатора, задаётся строкой до 31 символа.

Значение по умолчанию

Системное имя устройства (hosthame).

Необходимый уровень привилегий

15

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp chap hostname rtt1
```

ppp chap password

Данной командой указывается пароль, который отправляется удаленной стороне вместе с именем маршрутизатора для прохождения CHAP-аутентификации.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
ppp chap password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ppp chap password
```

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1 .. 64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

Пароль хранится в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp chap password ascii-text 01234567
```

ppp chap refuse

Данной командой включается игнорирование аутентификации.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
[no] ppp chap refuse
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Игнорирование аутентификации выключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp chap refuse
```

ppp chap username

Данной командой указывается пользователь для аутентификации удаленной стороны и осуществляется переход в режим конфигурирования пользователя.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Синтаксис

```
[no] ppp chap username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp chap username xap
```

ppp ipcp accept-address

Данной командой разрешается принимать от соседа любой ненулевой IP-адрес в качестве локального IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ppp ipcp accept-address
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Прием IP-адреса запрещен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp ipcp accept-address
```

ppp ipcp remote-address

Данной командой устанавливается IP-адрес, который отправляется удаленной стороне для последующего его присвоения.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленной стороны.

Синтаксис

```
ppp ipcp remote-address <ADDR>
no ppp ipcp remote-address
```

Параметры

<ADDR> – IP-адрес удаленного шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp ipcp remote-address 192.168.1.2
```

ppp max-configure

Данной командой устанавливается количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp max-configure <VALUE>
```

```
no ppp max-configure
```

Параметры

<VALUE> – время в секундах, принимает значения [1..255].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)#i ppp max-configure 4
```

ppp max-failure

Данной командой устанавливается количество попыток выслать Configure-NAK пакеты, прежде чем будут подтверждены все опции.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp max-failure <VALUE>
no ppp max-failure
```

Параметры

<VALUE> – время в секундах, принимает значения [1..255].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)#ppp max-failure 3
```

ppp max-terminate

Данной командой устанавливается количество попыток выслать Terminate-Request пакеты, прежде чем сессия будет прервана.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp max-terminate <VALUE>
no ppp max-terminate
```

Параметры

<VALUE> – время в секундах, принимает значения [1..255].

Значение по умолчанию

2

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp max-terminate 4
```

ppp mru

Данной командой указывается размер MRU (Maximum Receive Unit) для интерфейса.

Использование отрицательной формы команды (no) устанавливает значение MRU по умолчанию.

Синтаксис

```
ppp mru <MRU>
no ppp mru
```

Параметры

<MRU> – значение MRU, принимает значения в диапазоне [552..1500].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# mru 1400
```

ppp multilink

Данной командой включается режим MLPPP на E1-интерфейсе.

Использование отрицательной формы команды (no) отключает режим MLPPP.

Синтаксис

```
[no] ppp multilink
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-E1

Пример

```
rtt(config-if-e1)# ppp multilink
```

ppp multilink-group

Данной командой E1-интерфейс включается в группу агрегации.

Использование отрицательной формы команды (no) исключает интерфейс из группы агрегации.

Синтаксис

```
ppp multilink-group <GROUP-ID>
```

Параметры

<GROUP-ID> – идентификатор группы, принимает значение [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-E1

Пример

```
rtt(config-if-e1)# ppp multilink-group 1
```

ppp timeout keepalive

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp timeout keepalive [ <TIME >]
no ppp timeout keepalive
```

Параметры

<TIME> – время в секундах, принимает значения [1..32767].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp timeout keepalive 200
```

ppp timeout retry

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторяет запрос на установление сессии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ppp timeout retry <TIME>
no ppp timeout retry
```

Параметры

<TIME> – время в секундах, принимает значения [1..255].

Значение по умолчанию

3

Необходимый уровень привилегий

10

Командный режим

CONFIG-SERIAL

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-e1)# ppp timeout retry 3
```

show controllers e1

Данной командой выводится информация о E1-контролерах.

Синтаксис

```
show controllers e1 [<IF>]
```

Параметры

<IF> – имя физического интерфейса маршрутизатора, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show controllers e1
Interface 'te1/0/1':
 SFP present:     Yes
 SFP Vendor name: NSC-COM
 SFP Vendor PN:   611.900
 Line code:       HDB3
 Clock source:    Internal
 Timeslot:        24
 Invert Data:     No
 Framing CRC4:    No
 Loopback:        --
 CRC algorithm:   FCS16
 E1 Link:         Down
 E1 Synced:       No
 E1 RX AIS:       No
 E1 RX RAI:       No
```

switchport e1 clock source

Данной командой определяется источник синхронизации e1-интерфейса.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport e1 clock source { internal | line }
no switchport e1 clock source
```

Параметры

internal – используется внутренний источник синхронизации;

line – в качестве источника синхронизации используется сигнал с линии.

Значение по умолчанию

internal

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 clock source line
```

switchport e1 crc

Данной командой определяется режим проверки целостности передаваемых данных.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport e1 crc { 16 | 32 }
no switchport e1 crc
```

Параметры

16 – используется 16-битный алгоритм проверки отсутствия ошибок (CRC);

32 – используется 32-битный алгоритм проверки отсутствия ошибок (CRC).

Значение по умолчанию

16

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 crc 16
```

switchport e1 framing

Данной командой включается проверка целостности по алгоритму CRC4.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport e1 framing { crc4 | no-crc4 }
no switchport e1 framing
```

Параметры

crc4 – включение проверки целостности по CRC4;

no-crc4 – выключение проверки целостности по CRC4.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 framing crc4
```

switchport e1 invert data

Данной командой включается реверсивная отправка данных.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] switchport e1 invert data
```

Параметры

Отсутствуют.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 invert data
```

switchport e1 linecode

Данной командой определяется алгоритм кодирования данных, осуществляемое для их передачи по физическому каналу.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport e1 linecode { ami | hdb3 }
```

Параметры

ami – использовать алгоритм ami;

hdb3 – использовать алгоритм hdb3.

Значение по умолчанию

hdb3

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 linecode ami
```

switchport e1 slot

Данной командой порт e1 привязывается к физическому интерфейсу. Использование отрицательной формы команды (no) переходит в стандартный режим.

Синтаксис

```
[no] switchport e1 slot <SLOT>
```

Параметры

<SLOT> – идентификатор слота, принимает значение в диапазоне [0..12].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# switchport e1 0
```

switchport e1 timeslots

Данной командой определяется количество используемых 64 кбит/с каналов в потоке E1.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport e1 timeslots < RANGE >
no switchport e1 timeslots
```

Параметры

< RANGE > – количество 64 кбит/с каналов, принимает значение [1..31].

Значение по умолчанию

31

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 timeslots 16
```

switchport e1 unframed

Данной командой включается режим использования потока E1 как единого без разделения на каналы по 64 кбит/с.

Использование отрицательной формы командой (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] switchport e1 unframed
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt:rtt(config-if-gi)# switchport e1 unframed
```

switchport mode e1

Данной командой физический порт переводится в режим работы с SFP e1 модулем.

Использование отрицательной формы команды (no) переводит порт в стандартный режим.

Синтаксис

```
[no] switchport mode e1
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# switchport mode e1
```

12.5. Беспроводные модемы

allowed-auth

Данной командой задается метод аутентификации пользователя в мобильной сети.

Использование отрицательной формы команды (no) устанавливает метод аутентификации по умолчанию.

Синтаксис

```
allowed-auth <TYPE>
no allowed-auth
```

Параметры

<TYPE> – метод аутентификации пользователя в мобильной сети [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP].

Значение по умолчанию

PAP

Необходимый уровень привилегий

15

Командный режим

CONFIG-CELLULAR-PROFILE

Пример

```
rtt(config-cellular-profile)# allowed-auth MSCHAP
```

allowed-mode

Данной командой разрешается использование режима при работе USB-модема.

Использование отрицательной формы команды (no) удаляет разрешение на использование режима.

Синтаксис

```
[no] allowed-mode <MODE>
```

Параметры

<MODE> – допустимый режим работы USB-модема [2g, 3g, 4g]

Значение по умолчанию

Разрешены режимы, которые отображаются как "Allowed modes:" в выводе команды *show cellular status modem <ID>*.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# allowed-mode 4g
```

apn

Данной командой задается точка доступа мобильной сети.

Синтаксис

```
apn <NAME>
```

Параметры

<NAME> – точка доступа мобильной сети, задаётся строкой до 253 символов.

Значение по умолчанию

Не сконфигурирована.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-PROFILE

Пример

```
rtt(config-cellular-profile)# apn internet
```

cellular modem

Данной командой создается USB-модем с определенным идентификатором и осуществляется переход в режим конфигурирования USB-модема.

Использование отрицательной формы команды (no) удаляет сконфигурированный USB-модем.

Синтаксис

```
[no] cellular modem <ID>
```

Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# cellular modem 1
```

cellular profile

Данной командой создается профиль настроек для USB-модема с определенным идентификатором и осуществляется переход в режим конфигурирования профиля.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль настроек USB-модема.

Синтаксис

```
[no] cellular profile <ID>
```

Параметры

<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# cellular profile 1
```

device

Данной командой задается идентификатор USB-порта подключенного модема.

Использование отрицательной формы команды (no) удаляет установленный идентификатор.

Синтаксис

```
device <WORD>
no device
```

Параметры

<WORD> – идентификатор USB порта подключенного модема, задается строкой длиной от 1 до 12 символов.

Данный идентификатор должен совпадать с идентификатором в поле "USB port device" в выводе команды *show cellular status modem*.

Значение по умолчанию

Не указан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# device 1-1
```

enable

Данной командой активируется USB-модем.

Использование отрицательной формы команды (no) деактивирует USB-модем.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

USB-модем не активирован.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)#
```

ip-version

Данной командой ограничивается возможность использования семейств IP-адресов в мобильной сети.

Использование отрицательной формы команды (no) разрешает использование адресаций обоих семейств IPv4 и IPv6.

Синтаксис

```
ip-version { ipv4 | ipv6 }
no ip-version
```

Параметры

ipv4 – семейство IPv4;

ipv6 – семейство IPv6;

Значение по умолчанию

Разрешена адресация обоих семейств IPv4 и IPv6.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-PROFILE

Пример

```
rtt(config-cellular-profile)# ip-version ipv4
```

mode

Данной командой задается режим работы беспроводного модема.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы модема, принимает значения stick/hilink.

Значение по умолчанию

stick

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# mode hilink
```

mru

Данной командой задается размер максимального принимаемого пакета.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
mru { <MRU> }
no mru
```

Параметры

<MRU> – значение MRU, принимает значения в диапазоне [128..16383].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# mru 1476
```

number

Данной командой устанавливается номер дозвона для подключения к мобильной сети.

Использование отрицательной формы команды (no) удаляет номер дозвона для подключения к мобильной сети.

Синтаксис

```
number <WORD>
no number
```

Параметры

<WORD> – номер дозвона для подключения к мобильной сети, задаётся строкой от 1 до 15 символов.

Значение по умолчанию

Номер не задан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-PROFILE

Пример

```
rtt(config-cellular-profile)# number *99#
```

password

Команда для установки пароля пользователя мобильной сети в открытой или зашифрованной форме. Пароль пользователя хранится в конфигурации в зашифрованной форме. При конфигурировании можно задать пароль в открытой форме либо скопировать пароль в зашифрованной форме с другого устройства.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

Синтаксис

```
password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [1..64] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [2..128] символов.

Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPP-USER

Пример

```
rtt(config-ppp-user)# password ascii-text 01234567
```

pin

Данной командой задается код разблокировки SIM-карты.

Использование отрицательной формы команды (no) удаляет код разблокировки SIM-карты.

Синтаксис

```
pin <WORD>
no pin
```

Параметры

<WORD> – код разблокировки SIM-карты [4..8] символов. Возможно использование только цифр.

Значение по умолчанию

PIN не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# pin 4856
```

preferred-mode

Данной командой задается предпочтительный режим работы USB-модема в мобильной сети.

Использование отрицательной формы команды (no) удаляет предпочтительный режим работы USB-модема в мобильной сети.

Синтаксис

```
preferred-mode { <MODE> }
no preferred-mode
```

Параметры

<MODE> – предпочтительный режим работы USB-модема [2g, 3g, 4g].

Значение по умолчанию

Определяется USB-модемом.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# preferred-mode 4g
```

profile

Данной командой на USB-модем назначается созданный профиль настроек для USB-модема.

Использование отрицательной формы команды (no) удаляет назначенный профиль настроек для USB-модема.

Синтаксис

```
profile <ID>
no profile
```

Параметры

<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].

Значение по умолчанию

ID профиля настроек не задан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CELLULAR-MODEM

Пример

```
rtt(config-cellular-modem)# profile 1
```

show cellular configuration modem

Данной командой отображается информация о сконфигурированных USB-модемах.

Синтаксис

```
show cellular configuration modem [ <ID> ]
```

Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех сконфигурированных USB-модемах.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show cellular configuration modem
Number   State      Description                     USB port device   Profile
modem
------   --------   -------------------------------   ---------------   --------
1        Enabled    megafon                         1-1               1
2        Enabled    mts                             1-2               2

rtt# show cellular configuration modem 2
State:             Enabled
Description:       mts
USB port device:   1-2
Pin:               --
MRU:               --
MTU:               1500
Preferred mode:    none
Allowed modes:     all
Profile:           2
```

```
Description:          MTS
  User name:            mts
  Number:               *99#
  APN:                  internet.mts.ru
  Password (encrypted): 91A010
  IP version:           both
  Allowed auth:         EAP
Security zone:     --
```

show cellular configuration profile

Данной командой отображается информация о сконфигурированных профилях USB-модемов.

Синтаксис

```
show cellular configuration profile [ <ID> ]
```

Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех сконфигурированных профилях USB-модемов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show cellular configuration profile
Number    User name          APN                Number             Description
profile
-------   ----------------   ----------------   ----------------   ---------------
1         gdata              internet           *99#               MEGAFON
2         mts                internet.mts.ru    *99#               MTS

rtt# show cellular configuration profile 1
Description:          MEGAFON
User name:            gdata
Number:               *99#
APN:                  internet
Password (encrypted): 9BB00279B1
IP version:           both
Allowed auth:         EAP
```

show cellular status modem

Данной командой отображается информация о статусе подключенных USB-модемов.

Синтаксис

```
show cellular status modem [ <ID> ]
```

Параметры

<ID> – идентификатор USB-модема в системе [1..10].

Без указания номера USB-модема выводится таблица с краткой информацией о всех обнаруженных USB-модемах.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cellular status modem
Number   USB port   Manufacturer   Model     Current state   Inteface         Link
device                                                                       state
------   --------   ------------   -------   -------------   --------------   -----
1        1-1        huawei         E3372       disabled        --             Down

rtt# show cellular status modem 1
Interface 'modem 1' status information:
        USB port device:         1-1
        Manufacturer:            huawei
        Model:                   E3372
        Revision:                21.180.01.00.00
        IMEI:                    861821036192893
        Status SIM lock:         --
        Status unlock retries:   sim-pin (3) sim-pin2 (3) sim-puk (10) sim-puk2 (10)
        Current state:           disabled
        Access tech:             unknown
        Signal level:            0
        Support modes:
                allowed 2G; preferred none;
                allowed 3G; preferred none;
                allowed 4G; preferred none;
                allowed 2G 3G 4G; preferred none;
        Allowed modes:           2G 3G 4G
        Preferred modes:         none
        Type IP:                 IPv4
        Operator name:
        Registration:            unknown
```

user

Данной командой задается имя пользователя мобильной сети. После выполнения данной команды маршрутизатор переходит в режим конфигурирования параметров пользователя.

Использование отрицательной формы команды (no) удаляет имя пользователя мобильной сети.

Синтаксис

```
[no] user <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до от 1 до 31 символа.

Значение по умолчанию

Пользователь не создан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CELLULAR-PROFILE

Пример

```
rtt(config-cellular-profile)# user gdata
```

13.УПРАВЛЕНИЕ L2-ФУНКЦИЯМИ

13.1. Управление L2-функциями

bridge

Данной командой добавляется сетевой мост в систему и осуществляется переход в режим настройки его параметров.

Использование отрицательной формы команды (no) удаляет мост.

Синтаксис

```
[no] bridge <BRIDGE-ID>
```

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

Переход в режим конфигурирования сетевого моста *bridge 10*:

```
rtt(config)# bridge 10
rtt(config-bridge)#
```

bridge-group

Данная команда используется для добавления в текущего сетевого интерфейса в L2-домен.

Использование отрицательной формы команды (no) удаляет интерфейс из L2-домена.

Синтаксис

```
bridge-group <BRIDGE-ID> [tagged]
no bridge-group
```

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

tagged – ключ для направления тегированных кадров физического интерфейса в bridge. Применимо только на физических интерфейсах маршрутизаторов R100/200

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-L2TPV3

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-OPENVPN

Пример

```
rtt(CONFIG-IF-SUB)# bridge-group 15
```

description

Данная команда используется для назначения описания конфигурируемому сетевому мосту.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
```

```
no description
```

Параметры

<DESCRIPTION> – описание сетевого моста, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# description "broadway"
```

enable

Данной командой разрешается работа сетевого моста. Без указания данной команды сетевой мост не работает.

Использование отрицательной формы команды (no) отключает маршрутизацию данных.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# enable
```

mac-address

Данная команда позволяет задать MAC-адрес сетевого моста, физического или агрегированного интерфейса, отличного от системного.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mac-address <ADDR>
no mac-address
```

Параметры

<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Значение по умолчанию

Системный MAC-адрес.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# mac-address A8:F9:B0:00:00:04
```

ports vrrp filtering

Данной командой включается запрет на отправку VRRP-сообщений в интерфейсах, включенных в L2-домен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ports vrrp filtering {enable | exclude vlan}
```

Параметры

enable – включение запрета на отправку VRRP-сообщений в интерфейсах, включенных в L2-домен;

exclude vlan – при указании данного параметра VLAN исключается из списка фильтруемых.

Значение по умолчанию

Фильтрация VRRP-сообщений отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# ports vrrp filtering enable
```

protected-ports

Данной командой осуществляется управление режимом изоляции интерфейсов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
protected-ports <MODE>
```

Параметры

<MODE> – режим изоляции интерфейсов, принимает следующие значения:

- none – изоляция интерфейсов отключена. В данном режиме коммутация кадров между членами сетевого моста разрешена;
- local – изоляция интерфейсов включена. В данном режиме коммутация кадров между членами сетевого моста запрещена;
- radius – изоляция интерфейсов включена. Для использования данного режима требуется настройка Wi-Fi контроллера туннелей в режиме "radius". В данном режиме коммутация кадров между членами сетевого моста запрещена, за исключением SoftGRE DATA туннелей.
  Для SoftGRE DATA-туннелей параметры изоляции запрашиваются у RADIUS-сервера.

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# protected-ports local
```

protected-ports exclude

Данной командой исключаются их списка изолируемых сущности, включенные в сетевой мост. Данная опция актуальна при настройке режима protected-ports отличного от значения по умолчанию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] protected-ports exclude { <IF> | <TUN> | <VLAN> }
```

Параметры

<IF> – интерфейс, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля, задается в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

<VLAN> – vlan, связанный с сетевым мостом.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# protected-ports exclude vlan
```

show interfaces bridge

Данная команда используется для просмотра информации о VLAN, интерфейсах, туннелях, объединенных мостом.

Синтаксис

```
show interfaces bridge [<BRIDGE-ID>]
```

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces bridge 1
Bridges      Interfaces
----------   ----------------------------------------
bridge 1     vlan 1,gi1/0/1.10
```

thresholds wifi-tunnels-number high

Данная функция поддерживается только при наличии лицензии на Wi-Fi controller.

Данной командой устанавливается верхний порог аварии превышения числа включенных саб-туннелей softgre в bridge.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
thresholds wifi-tunnels-number high<NUM>
no thresholds wifi-tunnels-number high
```

Параметры

<NUM> – верхний порог аварии превышения числа включенных саб-туннелей softgre в bridge, принимает значение в диапазоне [1..1000].

Значение по умолчанию

950

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config)# thresholds wifi-tunnels-number high 200
```

thresholds wifi-tunnels-number low

Данная функция поддерживается только при наличии лицензии на Wi-Fi controller.

Данной командой устанавливается нижний порог аварии превышения числа включенных саб-туннелей softgre в bridge.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
thresholds wifi-tunnels-number low <NUM>
no thresholds wifi-tunnels-number low
```

Параметры

<NUM> – нижний порог аварии превышения числа включенных саб-туннелей softgre в bridge, принимает значение в диапазоне [1..1000].

Значение по умолчанию

925

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config)# thresholds wifi-tunnels-number low 20
```

unknown-unicast-forwarding disable

Данной командой запрещается коммутация unicast-трафика с отсутствующими MAC-адресами в таблице MAC-адресов сетевого моста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] unknown-unicast-forwarding disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Коммутация unicast-трафика с неизвестными MAC-адресами разрешена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# unknown-unicast-forwarding disable
```

vlan

Данная команда используется для связывания текущего сетевого моста с VLAN. Все порты, являющиеся членами назначаемого VLAN, автоматически включаются в сетевой мост и становятся участниками общего L2-домена. Для управления членством сетевых интерфейсов в VLAN используются команды, описанные в разделе Настройка и мониторинг VLAN.

Использование отрицательной формы команды (no) удаляет привязку VLAN и отключает соответствующие интерфейсы от сетевого моста.

Синтаксис

```
vlan <VID>
```

```
no vlan
```

Параметры

<VID> – идентификатор VLAN, задаётся в диапазоне [1..4095].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-vlan)# vlan 40
```

13.2. Управление Spanning Tree

errdisable recovery cause bpduguard

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данной командой включается автоматическое восстановление интерфейса после срабатывания BPDU Guard.

Синтаксис

[no] errdisable recovery cause bpduguard

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

```
rtt(config)# errdisable recovery cause bpduguard
```

errdisable recovery interval

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данной командой задаётся временной интервал, через который будет сделана попытка автоматического восстановления интерфейса после срабатывания BPDU Guard.

Синтаксис

[no] errdisable recovery cause <TIME>

Параметры

<TIME> – время в секундах, принимает значение в диапазоне [30…86400].

Значение по умолчанию

300

Необходимый уровень привилегий

10

Командный режим

CONFIG

```
rtt(config)# errdisable recovery interval 60
```

instance

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данной командой создается соответствие между экземпляром протокола MSTP и группами VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] instance <INSTANCE> vlan <VID>
```

Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<VID> – идентификационный номер VLAN, задаётся в диапазоне [1…4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-MST

Пример

```
rtt(config-mst)#instance 5 vlan 10-250
```

name

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данной командой указывается имя конфигурации MSTP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
name <NAME>
no name
```

Параметры

<NAME> – имя конфигурации MSTP, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-MST

Пример

```
rtt(config-mst)# name test
```

revision

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данной командой задается номер ревизии конфигурации MSTP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
revision <NUM>
no revision
```

Параметры

<NUM> – номер ревизии конфигурации MSTP, задается в диапазоне [0..65535].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-MST

Пример

```
rtt(config-mst)#revision 5000
```

set interface active

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данная команда восстанавливает интерфейс после срабатывания BPDU Guard.

Синтаксис

```
set interface active <IF>
```

Параметры

<IF> – физический или агрегированный интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# set interface active gigabitethernet 1/0/3
```

show errdisable recovery

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данная команда отображает информацию о настройке параметров восстановления после срабатывания BPDU Guard.

Синтаксис

```
show errdisable recovery
```

Параметры

Команда не содержит параметров

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show errdisable recovery
Timer interval: 300 seconds
Reason       Automatic Recovery
----------   ------------------
bpduguard    Disabled
```

show interfaces status errdisable

В текущей версии ПО данная команда не поддерживается на маршрутизаторах R800.

Данная команда отображает информацию о интерфейсах, которые были переведены в статус errdisable в результате срабатывания BPDU Guard.

Синтаксис

```
show interfaces status errdisable
```

Параметры

Команда не содержит параметров

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces status errdisable
Interface              Err-Disable Reason    Auto-Recovery Time Left
                                             (sec)
--------------------   -------------------   ------------------------
gi1/0/3                bpduguard             56
```

show spanning-tree

Данная команда отображает подробную информацию о настройке протокола семейства STP.

Синтаксис

```
show spanning-tree { <IF> | tunnel <TUN> | bpdu | bridge { vrf <VRF_NAME> | global
[active] | <BRIDGE_NUM> [ active ] [ vrf <VRF_NAME> ] } }
```

Параметры

<IF> – физический или агрегированный интерфейс или группа интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<BRIDGE> – номер сконфигурированного bridge;

<VRF_NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

global – при указании команды отображается информация о состоянии протокола STP/RSTP/MSTP;

active – при указании команды отображается информация о состоянии протокола STP/RSTP/MSTP, а также дополнительные сведения о состоянии интерфейсов, их типов, ролей, стоимости и приоритета;

bpdu – при указании команды отображается информация о работе устройства с BPDU: filtering – пакеты BPDU будут отброшены, flooding – пакеты BPDU будут рассылаться в широковещательный домен;

bridge – при указании команды отображается информация о состоянии протокола VSTP для соответствующего bridge-домена.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show spanning-tree gigabitethernet 1/0/10
Port gi1/0/10 disabled
State: BLK
Port id: ---
Type: ---
Designated bridge Priority: ---
Designated port id: ---
Role: ---
Port cost: ---
Designated path cost: ---
Address: ---
Port Fast: ---
rtt# show spanning-tree bridge
Protocol version: STP
     Root ID: [32768] 02:01:02:03:04:55
              Root port: [128] gigabitethernet 1/0/14
              Pathcost 4
              Message Age 1
              Hello time: 2 Max age time: 20 Forward delay: 15
   Bridge ID: [32768] 02:20:03:A0:04:90
              Hello time: 2 Max age time: 20 Forward delay: 15
              Transmit hold count: 6 Topology change: 0
              Time since topology change: 13736 Topology change count: 2 show
```

spanning-tree

Команда активирует работу протоколов семейства Spanning Tree (STP, RSTP, MSTP).

Протокол MSTP поддерживается только на маршрутизаторе R800.

Использование отрицательной формы команды (no) выключает поддержку протоколов семейства Spanning Tree.

Синтаксис

```
[no] spanning-tree
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# spanning-tree
```

spanning-tree

Команда активирует работу протокола VSTP для широковещательного домена, организованного с помощью bridge.

Использование отрицательной формы команды (no) выключает поддержку протокола VSTP для широковещательного домена, организованного с помощью bridge.

Синтаксис

```
[no] spanning-tree
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# spanning-tree
```

spanning-tree bpdu

Данной командой определяется режим обработки пакетов BPDU интерфейсом, на котором выключен протокол STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree bpdu <MODE>
no spanning-tree bpdu
```

Параметры

<MODE> – режим работы:

- filtering – на интерфейсе с выключенным протоколом STP BPDU-пакеты фильтруются;
- flooding – на интерфейсе с выключенным протоколом STP нетегированные BPDU-пакеты передаются, тегированные – фильтруются.

Значение по умолчанию

flooding

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree bpdu filtering
```

spanning-tree bpduguard

Данной командой включается функционал BPDU Guard на интерфейсе.

Синтаксис

```
[no] spanning-tree bpduguard
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN-SERVER

CONFIG-OPENVPN

Пример

```
rtt(config-if-gi)# spanning-tree bpduguard
```

spanning-tree cost

Данной командой устанавливается размер, на который увеличивается поле cost при коммутации DPDU-пакета.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree cost <COST>
no spanning-tree cost
```

Параметры

<COST> – стоимость пути, устанавливается в диапазоне [1..20000000].

Значение по умолчанию

4

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree cost 115
```

spanning-tree disable

Данной командой запрещается работа протокола STP на конфигурируемом интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] spanning-tree disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

STP на интерфейсе включен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree disable
```

spanning-tree forward-time

Данной командой устанавливается интервал времени, затрачиваемый на прослушивание и изучение состояний перед переключением в состояние передачи.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree forward-time <TIME>
no spanning-tree forward-time
```

Параметры

<TIME> – время в секундах, принимает значения [4..30].

Значение по умолчанию

15

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-BRIDGE

Пример

```
rtt(config)# spanning-tree forward-time 20
```

spanning-tree guard root

Данной командой включается блокировка получения BPDU с более высоким STP-приоритетом.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] spanning-tree guard root
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree guard root
```

spanning-tree hello-time

Данной командой устанавливается интервал времени между отправкой BPDU-пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree hello-time <TIME>
no spanning-tree hello-time
```

Параметры

<TIME> – время в секундах, принимает значения [1..10].

Значение по умолчанию

2

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-BRIDGE

Пример

```
rtt(config)# spanning-tree hello-time 20
```

spanning-tree link-type

Данной командой устанавливается протокол RSTP в передающее состояние и определяет тип связи для выбранного порта – «точка-точка», «разветвлённый».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree link-type { point-to-point | shared }
no spanning-tree link-type
```

Параметры

point-to-point – команда определяет интерфейс как «точка-точка»;

shared – команда определяет интерфейс как «разветвленный».

Значение по умолчанию

point-to-point

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree link-type point-to-point
```

spanning-tree max-age

Данной командой устанавливается время жизни связующего дерева STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree max-age <TIME>
no spanning-tree max-age
```

Параметры

<TIME> – время в секундах, принимает значения [6..40].

Значение по умолчанию

20

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-BRIDGE

Пример

```
rtt(config)# spanning-tree max-age 35
```

spanning-tree mode

Данной командой выбирается поддерживаемый протокол из семейства STP для широковещательного домена, организованного с помощью bridge или физических интерфейсов маршрутизатора в режиме switchport.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree mode <MODE>
no spanning-tree mode
```

Параметры

<MODE> – протокол семейства STP:

- STP – IEEE 802.1D Spanning Tree Protocol;
- RSTP – IEEE 802.1W Rapid Spanning Tree Protocol;
- MSTP – IEEE 802.1s Multiple Spanning Trees (доступно только в глобальной конфигурации на R800).

Значение по умолчанию

RSTP

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-BRIDGE

Пример

```
rtt(config)# spanning-tree mode STP
```

spanning-tree mst configuration

В текущей версии ПО данная команда поддерживается только на маршрутизаторе R800.

Данной командой осуществляется переход в режим конфигурирования MSTP-параметров.

Синтаксис

```
spanning-tree mst configuration
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# spanning-tree mst configuration
rtt(config-mst)#
```

spanning-tree mst cost

В текущей версии ПО данная команда поддерживается только на маршрутизаторе R800.

Данной командой устанавливается метод определения ценности пути для экземпляра MST.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree mst <INSTANCE> cost <COST>
no spanning-tree mst <INSTANCE>
```

Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<COST> – стоимость пути, устанавливается в диапазоне [1..20000000].

Значение по умолчанию

4

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-TE

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree mst 1 cost 115
```

spanning-tree mst max-hops

В текущей версии ПО данная команда поддерживается только на маршрутизаторе R800.

Данной командой устанавливается максимальное количество транзитных участков для пакета BPDU, необходимых для формирования дерева и удержания информации о его строении. Если пакет уже прошел максимальное количество транзитных участков, то на следующем участке он отбрасывается.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree mst max-hops <NUM>
no spanning-tree mst max-hops
```

Параметры

<NUM> – количество транзитных участков, задается в диапазоне [6..40].

Значение по умолчанию

20

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# spanning-tree mst max-hops 10
```

spanning-tree mst port-priority

В текущей версии ПО данная команда поддерживается только на маршрутизаторе R800.

Данной командой устанавливается приоритет интерфейса для экземпляра MST.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree mst <INSTANCE> port-priority <PRIORITY>
```

```
no spanning-tree mst <INSTANCE> port-priority
```

Параметры

<INSTANCE> – идентификатор MST-экземпляра, указывается в диапазоне [1..15];

<PRIORITY> – приоритет, указывается в диапазоне с шагом 16 [0..240].

Значение по умолчанию

128

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# spanning-tree mst 1 port-priority 160
```

spanning-tree pathcost method

Данной командой устанавливается метод определения ценности пути.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree pathcost method { short | long }
no spanning-tree pathcost method
```

Параметры

long – значение ценности в диапазоне [1..200000000];

short – значение ценности в диапазоне [1..65535].

Значение по умолчанию

short

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# spanning-tree pathcost method short
```

spanning-tree portfast

Данной командой включается режим, в котором L2-интерфейс при поднятии на нем линка сразу переходит в состояние передачи, не дожидаясь истечения таймера.

Использование отрицательной формы команды (no) выключает режим моментального перехода в состояние передачи по поднятию линка.

Синтаксис

```
[no] spanning-tree portfast
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree portfast
```

spanning-tree port-priority

Данной командой устанавливается приоритет интерфейса в связующем дереве STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree port-priority <PRIORITY>
no spanning-tree port-priority
```

Параметры

<PRIORITY> – приоритет, указывается в диапазоне с шагом 16 [0..240].

Значение по умолчанию

128

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TPV3

CONFIG-OPENVPN

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-if-gi)# spanning-tree port-priority 160
```

spanning-tree priority

Данной командой настраивается приоритет связующего дерева STP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
spanning-tree priority <PRIORITY>
no spanning-tree priority
```

Параметры

<PRIORITY> – приоритет, указывается в диапазоне с шагом 4096 [0..61440].

Значение по умолчанию

32768

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-BRIDGE

Пример

```
rtt(config)# spanning-tree priority 4096
```

13.3. Настройка и мониторинг VLAN

force-up

Данная команда используется для включения режима активности VLAN вне зависимости от состояния интерфейсов, на которых разрешена обработка фреймов данного VLAN.

Использование отрицательной формы команды (no) отменяет режим активности VLAN.

Синтаксис

```
[no] force-up
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VLAN

Пример

```
rtt(config-vlan)# force-up
```

ip internal-usage-vlan

В текущей версии данная команда поддерживается только на маршрутизаторах R800.

Данная команда используется для резервирования VLAN для внутреннего использования на интерфейсе.

Использование отрицательной формы команды (no) отменяет резервирование.

Синтаксис

```
ip internal-usage-vlan <VID>
no ip internal-usage-vlan
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# ip internal-usage-vlan 1500
```

name

Данная команда используется для добавления описания VLAN.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
name <NAME>
no name
```

Параметры

<NAME> – описание VLAN, задаётся строкой до 255 символов.

Значение по умолчанию

Описание не задано.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VLAN

Пример

```
rtt(config)# name L2-ACCESS
```

show interfaces switch-port vlans

Данная команда используется для просмотра режима участия интерфейсов в VLAN.

Синтаксис

```
show interfaces switch-port vlans [<IF>]
```

Параметры

<IF> – имя физического или агрегированного интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будет отображена информация о всех интерфейсах заданной группы. При выполнении команды без параметра будет показана информация для всех физических интерфейсов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces switch-port vlans gigabitethernet 1/0/1-7
Interface   PVID   Frame types     Ingress     Tagged        Untagged
                                   filtering
---------   ----   -------------   ---------   -----------   -----------
gi1/0/1     1      All             yes         101           1
gi1/0/2     1      All             yes         150-151       1
gi1/0/3     1      All             yes         none          1
gi1/0/4     1      All             yes         none          1
gi1/0/5     1      All             yes         55            1
gi1/0/6     1      All             yes         none          1
gi1/0/7     1      All             yes         none          1
N/A - interface doesn't exist
N/S - interface is not a 802.1Q bridge port
ERR - can't get vlan setting for interface
```

show vlans

Данная команда используется для просмотра информации об определенной VLAN.

Синтаксис

```
show vlans [<VID>]
```

Параметры

<VID> – идентификационный номер VLAN, диапазон допустимых значений [1 .. 4094].

Можно указать несколько VLAN перечислением через запятую «,» либо указать диапазон VLAN через дефис «-». При выполнении команды без параметра будут показаны все созданные VLAN.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show vlans
VID    Name                   Tagged                 Untagged
----   --------------------   --------------------   ----------------------
1      default                                       gi1/0/3-4, gi1/0/6-24,
                                                     po1
2      --                                            gi1/0/1, te1/0/1-2
```

show vlans internal-usage

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данная команда используется для просмотра информации по VLAN, используемых системой.

Синтаксис

```
show vlans internal-usage
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show vlans internal-usage
Usage      VID    Reserved   IP address
--------   ----   --------   ----------
gi1/0/18   4088   No         Active
gi1/0/16   4089   No         Active
gi1/0/15   4090   No         Active
```

switchport access vlan

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R100/200.

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы access.

Синтаксис

```
switchport access vlan <VID>
no switchport access vlan
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport access vlan 50
```

switchport default-vlan tagged

Данная команда используется для изменения членства интерфейса во VLAN по умолчанию на тегированное.

Использование отрицательной формы команды (no) изменяет членство интерфейса во VLAN по умолчанию на нетегированное.

Синтаксис

```
[no] switchport default-vlan tagged
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport default-vlan tagged
```

switchport forbidden default-vlan

Данная команда используется для удаления интерфейса из VLAN по умолчанию.

Использование отрицательной формы команды (no) разрешает добавление vlan на порту.

Синтаксис

```
[no] switchport forbidden default-vlan
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# no switchport forbidden default-vlan
```

switchport general acceptable-frame-type

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой задается тип фреймов, которые может принимать интерфейс.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport general acceptable-frame-type { tagged-only | all }
no switchport general acceptable-frame-type
```

Параметры

tagged-only – принимать только тегированные фреймы;

all – принимать все фреймы.

Значение по умолчанию

all

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config)# switchport general acceptable-frame-type tagged-only
```

switchport general allowed vlan

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данная команда используется для включения/исключения интерфейса в/из VLAN.

Синтаксис

```
switchport general allowed vlan <ACT> <VID> [<TYPE>]
```

Параметры

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094]. Можно задать диапазоном через «-» или перечислением через «,»;

<TYPE> – тип пакета:

- tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;
- untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными. VLAN, в которую будут направлены входящие нетегированные пакеты, настраивается командой *switchport general pvid*, описанной в switchport general pvid.

Значение по умолчанию

Если не указывать параметр <TYPE>, то по умолчанию устанавливается «tagged».

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример 1

Исключить интерфейс из членства в VLAN 50:

```
rtt(config-if-gi)# switchport general allowed vlan remove 50
```

Пример 2

Включить интерфейс в VLAN 10-50 как тегированный:

```
rtt(config-if-gi)# switchport general allowed vlan add 10-50
```

switchport general allowed vlan auto-all

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данная команда включает автоматическое добавление интерфейса во все созданные на маршрутизаторе VLAN.

Использование отрицательной формы команды (no) отключает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

Синтаксис

```
[no] switchport general allowed vlan auto-all [ <TYPE> ]
```

Параметры

<TYPE> – тип пакета:

- tagged – интерфейс будет передавать и принимать пакеты в указанных VLAN тегированными;
- untagged – интерфейс будет передавать пакеты в указанных VLAN нетегированными. VLAN, в которую будут направлены входящие нетегированные пакеты, настраивается командой *switchport general pvid*, описанной в switchport general pvid.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport general allowed vlan auto-all
```

switchport general ingress-filtering disable

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данная команда используется для выключения фильтрации входящих пакетов на основе присвоенного им значения VLAN ID.

Использование отрицательной формы команды (no) включает фильтрацию.

Синтаксис

```
switchport general ingress-filtering disable
no switchport general ingress-filtering
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Фильтрация включена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport general ingress-filtering disable
```

switchport general macs-group

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой назначается профиль MAC-адресов и сопоставляется VLAN-ID, в который будут попадать пакеты с MAC-адресом источника, являющегося частью профиля MAC-адресов. Информация о конфигурировании профилей находится в разделе Управление профилями.

Использование отрицательной формы команды (no) удаляет сопоставление профиля MAC-адресов и VLAN-ID.

Синтаксис

```
switchport general macs-group <NAME> vlan <VID>
no switchport general macs-group <NAME>
```

Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа;

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2..4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport general macs-group OGM1 vlan 999
```

switchport general pvid

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой устанавливается идентификатор VLAN-порта (PVID) для входящего нетегированного трафика.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
switchport general pvid <VID>
no switchport general pvid
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [1…4094].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-FO

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport general pvid 999
```

switchport macs-group

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R100/200.

Данной командой назначается профиль MAC-адресов и сопоставляется VLAN-ID, в который будут попадать пакеты с MAC-адресом источника, являющегося частью профиля MAC-адресов. Информация о конфигурировании профилей находится в разделе Управление профилями.

Использование отрицательной формы команды (no) удаляет сопоставление профиля MAC-адресов и VLAN-ID.

Синтаксис

```
switchport macs-group <NAME> vlan <VID>
no switchport macs-group <NAME>
```

Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа;

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2..4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport macs-group OGM1 vlan 999
```

switchport trunk allowed vlan

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R100/200.

Данная команда используется для включения/исключения интерфейса в/из VLAN в режиме работы trunk.

Синтаксис

```
switchport trunk allowed vlan <ACT> <VID>
```

Параметры

<ACT> – назначаемое действие:

- add – включение интерфейса во VLAN;
- remove – исключение интерфейса из VLAN.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094]. Можно задать диапазоном через «-» или перечислением через «,».

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport trunk allowed vlan add 10-50
```

switchport trunk allowed vlan auto-all

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R100/200.

Данная команда включает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

Использование отрицательной формы команды (no) отключает автоматическое добавление порта во все созданные на маршрутизаторе VLAN.

Синтаксис

```
[no] switchport trunk allowed vlan auto-all
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport trunk allowed vlan auto-all
```

switchport trunk native-vlan

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R100/200.

Данная команда используется для настройки VLAN по умолчанию для интерфейса в режиме работы trunk. Весь нетегированный трафик, поступающий на данный интерфейс, направляется в данную VLAN.

Синтаксис

```
switchport trunk native-vlan <VID>
```

```
no switchport trunk native-vlan
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# switchport trunk native-vlan 55
```

vlan

Данной командой добавляется VLAN в систему и осуществляется переход в режим настройки параметров VLAN. На маршрутизаторе всегда существует VLAN с идентификатором 1, все интерфейсы по умолчанию включены в данный VLAN.

Использование отрицательной формы команды (no) удаляет VLAN.

Синтаксис

```
[no] vlan <VID>
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# vlan 40
```

13.4. Настройка и мониторинг Voice-VLAN

application

Данной командой настраивается приложение, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет настройку приложения для политики.

Синтаксис

```
application <APP-TYPE>
no application
```

Параметры

<APP-TYPE> – тип приложения, для которого будет срабатывать network-policy. Принимает значения:

- voice;
- voice-signaling;
- guest-voice;
- guest-voice-signaling;
- softphone-voice;
- video-conferencing;
- streaming-video;
- video-signaling.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-NET-POLICY

Пример

```
rtt(config-net-policy)# application voice
```

dscp

Данной командой настраивается значение DSCP, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет рассылку DSCP для данной сетевой политики (network-policy).

Синтаксис

```
dscp <DSCP>
dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

Рассылка DSCP в LLDPDU-MED отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-NET-POLICY

Пример

```
rtt(config-net-policy)# dscp 62
```

lldp network-policy

Данной командой на интерфейс назначается созданная ранее сетевая политика (network-policy).

Использование отрицательной формы команды (no) снимает назначенную ранее сетевую политику с интерфейса.

LLDPDU-сообщения, описанные политикой, будут рассылаться только если в глобальной конфигурации включена поддержка LLDP-MED (см. команду **lldp med fast-start enable** в разделе Настройка протокола LLDP).

Синтаксис

```
[no] lldp network-policy <NAME>
```

Параметры

<NAME> – имя назначаемой политики, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi) lldp network-policy ip-phones
```

network-policy

Данной командой создается сетевая политика (network-policy) и осуществляется переход в режим настройки параметров политики.

Использование отрицательной формы команды (no) удаляет созданную ранее сетевую политику (network-policy).

Синтаксис

```
[no] network-policy <NAME>
```

Параметры

<NAME> – имя создаваемой политики, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# network-policy ip-phones
```

priority

Данной командой настраивается значение COS, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет рассылку COS для данной сетевой политики (network-policy).

Синтаксис

```
priority <COS>
no priority
```

Параметры

<COS> – значение приоритета, принимает значения:

- best-effort – COS0;
- background – COS1;
- excellent-effort – COS2;
- critical-applications – COS3;
- video – COS4;
- voice – COS5;
- internetwork-control – COS6;
- network-control – COS7.

Значение по умолчанию

Рассылка COS в LLDPDU-MED отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-NET-POLICY

Пример

```
rtt(config-net-policy)# priority voice
```

vlan

Данной командой настраивается значение VLAN, которое будет числиться в пакете LLDP-MED при рассылке LLDPDU с интерфейсов, на которых назначена сетевая политика (network-policy).

Использование отрицательной формы команды (no) удаляет конфигурацию VLAN для данной сетевой политики (network-policy).

Синтаксис

```
vlan <VID> [tagged]
```

Параметры

<VID> – идентификационный номер VLAN, принимает значения [1…4094];

tagged – ключ, при установке которого абонентское устройство будет отправлять Ethernet-фреймы указанного приложения в тегированном виде.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-NET-POLICY

Пример

```
rtt(config-net-policy)# vlan 3258 tagged
```

14.РАБОТА С АДРЕСНЫМИ ТАБЛИЦАМИ

clear arp-cache

Команда используется для очистки содержимого ARP-таблицы.

Синтаксис

```
clear arp-cache [ <OPTIONS> ]
```

Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. Опциональный параметр, при указании которого будет очищена ARP-таблица в указанном VRF;
- <IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;
- ip-address <ADDR> – IP-адрес, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
- mac-address <ADDR> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear arp-cache ip-address 10.0.0.8
```

clear ipv6 neighbors

Команда используется для очистки содержимого IPv6 Neighbor Discovery таблицы.

Синтаксис

```
clear ipv6 neighbors [<OPTIONS> ]
```

Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет очищена IPv6 Neighbor Discovery таблица в указанном VRF;
- <IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;
- ipv6-address <IPV6-ADDR> – указывается IPv6-адрес, по которому ведется поиск, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
- mac-address <ADDR> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ipv6 neighbors
```

clear mac address-table

Команда используется для удаления информации об изученных MAC-адресах.

Синтаксис

```
clear mac address-table [ { dynamic | static } ] [ { interface { <IF> | host-port
<U/S/P> } | bridge <BRIDGE-ID> | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> }
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<U/S/P> – юнит (1), слот (0) и номер интерфейса пакетного процессора;

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<ID> – идентификатор туннеля;

<VLAN-ID> – номер vlan. Возможно задание списка vlan через символ "," без пробелов, диапазона vlan через символ "-" и/или комбинации списков и диапазонов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear mac address-table
```

ip arp

Данной командой добавляется статическая запись в ARP-таблицу.

Использование отрицательной формы команды (no) удаляет статическую запись из ARP-таблицы.

Синтаксис

```
ip arp [ vrf <VRF> ] <IP> <MAC> { <IF> | <TUN> }
no ip arp [ vrf <VRF> ] <IP>
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IP> – IP-адрес статической записи, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip arp 192.168.54.22 a8:f9:4b:ab:2e:d0 bridge 3
```

ip arp inspection

Данной командой включается функция контроля протокола ARP (ARP Inspection), предназначенная для защиты от атак с использованием протокола ARP (например, ARP-spoofing — перехват ARP-трафика).

С включенной функцией контроля ARP при приеме ARP-ответа производится сравнение старого и нового MAC-адресов, и при обнаружении его изменения запускается процедура верификации. Посылается ARP-запрос, требующий всем хозяевам IP-адреса сообщить свои MAC-адреса. Если выполняется атака, настоящая система, имеющая этот IP-адрес, ответит на запрос, и таким образом атака будет распознана. Если же изменение MAC-адреса было связано не с атакой, а со стандартными ситуациями, ответа, содержащего "старый" MAC-адрес, не будет, и по прошествии определенного таймаута система обновит запись в кэше.

Использование отрицательной формы команды (no) выключает функцию контроля.

Синтаксис

```
[no] ip arp inspection
```

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip arp inspection
```

ip arp reachable-time

Данной командой устанавливается время жизни записи в ARP-таблице.

Использование отрицательной формы команды (no) устанавливает значение параметра arp reachable-time по умолчанию.

Синтаксис

```
ip arp reachable-time <TIME>
no ip arp reachable-time
```

Параметры

< TIME > – время жизни динамических MAC-адресов, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]*< TIME >.

Необходимый уровень привилегий

10

Значение по умолчанию

160000

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-LOOPBACK

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip arp reachable-time 6000
```

ipv6 nd

Данной командой добавляется статическая запись в ND-таблицу.

Использование отрицательной формы команды (no) удаляет статическую запись из ND-таблицы.

Синтаксис

```
ipv6 nd [ vrf <VRF> ] <IPV6> <MAC> {<IF> | <TUN>}
no ipv6 nd [ vrf <VRF> ] <IP>
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IPV6-ADDR> – указывается IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 a8:f9:4b:ab:2e:d0
```

ipv6 nd reachable-time

Данной командой устанавливается время, в течение которого удаленный узел IPv6 считается доступным при отсутствии активности узла.

Использование отрицательной формы команды (no) устанавливает значение параметра nd reachable-time по умолчанию.

Синтаксис

```
ipv6 nd reachable-time <TIME>
```

```
no ipv6 nd arp reachable-time
```

Параметры

<TIME> – время жизни записи об удаленном узле IPv6 в таблице ND протокола, в миллисекундах. Допустимые значения от 5000 до 100000000 миллисекунд. Реальное время обновления записи варьируется от [0,5;1,5]*< TIME >.

Значение по умолчанию

30000

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 nd reachable-time 27000
```

logging firewall arp-inspection

Команда включает режим логирования атак с использованием протокола ARP, работает только совместно с командой **ip arp inspection**.

При включенных командах **ip arp inspection** и **logging firewall arp-inspection** в случае обнаружения атаки типа ARP-Spoofing в Syslog будет отправлено сообщение с указанием интерфейса существующей и подменяемой arp-записях.

Использование отрицательной формы команды (no) отключает режим логирования.

Синтаксис

```
[no] logging firewall arp-inspection
```

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# logging firewall arp-inspection

2024-08-13T12:29:01+07:00 %FIREWALL-W-WARN: arp-inspection
a8:f9:4b:ac:4d:3e(192.168.1.10) -> a8:f9:4b:aa:cc:a4(192.168.1.11) denied (gi1/0/1
08:00:27:d2:26:79)
```

mac address-table aging-time

Командой устанавливается время жизни динамических MAC-адресов в forwarding-таблице.

Использование отрицательной формы команды (no) устанавливает значение «aging time» по умолчанию.

Синтаксис

```
mac address-table aging-time <AGING TIME>
[no] mac address-table aging time
```

Параметры

<AGING TIME> – время жизни динамических MAC-адресов, в секундах. Допустимые значения:

- R800 – от 10 до 630 секунд. При значении 0 таймер выключен.
- R100/200 – от 20 до 630 секунд.

Значение по умолчанию

300

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# mac address-table aging-time 30
```

mac address-table save-secure-freq

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой устанавливается частота сохранения списка статических (secure) MAC-адресов.

Использование отрицательной формы команды (no) устанавливает значение «mac address-table save-secure-freq» по умолчанию.

Синтаксис

```
mac address-table save-secure-freq <SAVE-SECURE-FREQ>
[no] mac address-table save-secure-freq
```

Параметры

<SAVE-SECURE-FREQ> – частота сохранения списка статических (secure) MAC-адресов, принимает значение [600..86400] секунд.

Значение по умолчанию

1200 секунд

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# mac address-table save-secure-freq 650
```

port-security max

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой устанавливается максимальное количество MAC-адресов, разрешенных для запоминания на порту.

Использование отрицательной формы команды (no) отключает «port-security».

Синтаксис

```
port-security max <MAX>
no port-security max
```

Параметры

<MAX> – максимальное количество MAC-адресов, которое будет запоминаться портом, принимает значения [1..1024].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-TE

Пример

```
rtt(config-if-gi)# port-security max 1
```

port-security mode

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данная команда позволяет настроить режим «port-security».

Использование отрицательной формы команды (no) отключает режим безопасности.

Синтаксис

```
port-security mode [<OPTIONS>]
no port-security mode
```

Параметры

<OPTIONS> – параметры команды для выбора режима port-security:

limited – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса не сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

lock – при включении данного режима:

- на порту сохраняются все выученные MAC-адреса;
- порт не запоминает новые адреса;
- MAC-адреса сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

secure-delete-on-reset – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса не сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов не зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

secure-permanent – при включении данного режима:

- с порта удаляются все выученные MAC-адреса;
- количество адресов, которое снова может запомнить порт, ограничивается текущей конфигурацией;
- MAC-адреса сохраняются между аппаратными перезагрузками;
- время хранения MAC-адресов не зависит от времени жизни динамических MAC-адресов в forwarding-таблице.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-TE

Пример

```
rtt(config-if-gi)# port-security mode secure-delete-on-reset
rtt(config-if-gi)# port-security mode secure-permanent
```

port-security unknown-sa-action

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Командой устанавливается запрет на передачу пакетов с неизвестными MAC-адресами.

Использование отрицательной формы команды (no) разрешает передачу пакетов с неизвестными MAC-адресами.

Синтаксис

```
port-security unknown-sa-action discard
no port-security unknown-sa-action
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-TE

Пример

```
rtt(config-if-gi)# port-security unknown-sa-action discard
```

show arp

Команда используется для просмотра ARP-таблицы.

Синтаксис

```
show arp [<OPTIONS>]
```

Параметры

<options> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена ARP-таблица в указанном VRF;

- <IF> – наименование системного интерфейса или списка интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора. Отображается только информация по указанным интерфейсам;
- <TUN> – наименования туннелей, задаются в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.
- mac-address <MAC> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- ip-address <ADDR> – IP-адрес, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show arp
Interface          IP address        MAC address          State             Age(min)
---------------    ---------------   -----------------    ---------------   ---------
bridge 1           192.168.1.1       a8:f9:4b:aa:00:40    --                --
gi1/0/5            10.255.100.1      d8:50:e6:d2:f0:46    reachable         2
gi1/0/5            10.255.100.5      a8:f9:4b:aa:00:45    --                --
```

show arp configuration

Команда используется для просмотра значений времени жизни записей в ARP-таблице.

Синтаксис

```
show arp configuration <IF>
```

Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh arp configuration gigabitethernet 1/0/1-5
Globally configured ARP reachable time is 6000 msec
Interface          ARP reachable time, msec
---------------    ------------------------
gi1/0/1            6000
gi1/0/2            6000
gi1/0/3            6000
gi1/0/4            6000
gi1/0/4            6000
```

show ipv6 neighbors

Команда используется для просмотра IPv6 Neighbor Discovery таблицы.

Синтаксис

```
show ipv6 neighbors [<OPTIONS>]
```

Параметры

<OPTIONS> – параметры команды для детализации запрашиваемой информации, опциональный параметр:

- vrf <VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будет отображена IPv6 Neighbor Discovery таблица в указанном VRF;
- <IF> – наименование системного интерфейса или списка интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора. Отображается только информация по указанным интерфейсам;
- mac-address <MAC> – указывается MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- ipv6-address <IPV6-ADDR> – указывается IPv6-адрес, по которому ведется поиск, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 neighbors
Interface          IPv6 address                MAC address          State       Age(min)
---------------    -------------------------   -----------------    -----------   ------
gi1/0/5            fc00::1                     d8:50:e6:d2:f0:46    reachable       1
gi1/0/5            fc00::2                     a8:f9:4b:aa:00:45    --              --
bridge 1           fe80::aaf9:4bff:feaa:40     a8:f9:4b:aa:00:40    --              --
bridge 2           fe80::aaf9:4bff:feaa:40     a8:f9:4b:aa:00:40    --              --
gi1/0/5            fe80::aaf9:4bff:feaa:45     a8:f9:4b:aa:00:45    --              --
gi1/0/5            ff02::16                    33:33:00:00:00:16    norarp          --
gi1/0/5            ff02::fb                    33:33:00:00:00:fb    norarp          --
gi1/0/5            ff02::1:ff00:1              33:33:ff:00:00:01    norarp          --
```

```
gi1/0/5          ff02::1:ff00:2          33:33:ff:00:00:02   norarp          --
```

show ipv6 neighbors configuration

Команда используется для просмотра значений времени жизни записи об удаленном узле в таблице ND протокола.

Синтаксис

```
show ipv6 neighbors configuration <IF>
```

Параметры

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh ipv6 neighbors configuration tengigabitethernet 1/0/1-2
Globally configured NDP reachable time is 30000 msec
Interface         ND reachable time, msec
---------------   -------------------------
te1/0/1           30000
te1/0/2           30000
```

show mac address-table

Команда используется для просмотра информации, находящейся в таблице MAC-адресов.

Синтаксис

```
show mac address-table [ { count { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> } | {
static | dynamic} { bridge <BRIDGE-ID> | l2vpn p2p <P2P-NAME> | interface { <IF> |
host-port <U/S/P> } | tunnel { gre | softgre } <ID> | vlan <VLAN-ID> | mac <MAC-
ADDR> <MAC-MASK> } } ]
```

Параметры

- count – просмотр количества записей в таблице MAC-адресов. Список MAC-адресов не отображается;
- static – просмотр записей в таблице MAC-адресов внесенных из конфигурации маршрутизатора;
- dynamic – просмотр записей в таблице MAC-адресов внесенных в процессе изучения MAC-адресов;

- <BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;
- <P2P-NAME> – имя существующего p2p-сервиса, задается строкой до 31 символа;
- <IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;
- <HOST-U/S/P> – интерфейс пакетного процессора. Указывается в виде юнит (1), слот (0) и номер интерфейса пакетного процессора;
- <ID> – идентификатор туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;
- <VLAN-ID> – номер vlan. Возможно задание списка vlan через символ "," без пробелов, диапазона vlan через символ "-" и/или комбинации списков и диапазонов;
- <MAC-ADDR> – MAC-адрес, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
- [MAC-MASK] – маска MAC-адреса, опциональный параметр, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске. Значение маски по умолчанию FF:FF:FF:FF:FF:FF.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show mac address-table
VID     MAC Address          Interface                      Type
-----   ------------------   ----------------------------   -------
102     a8:f9:4b:aa:44:bb    host-port 1/0/2                Dynamic
101     a8:f9:4b:aa:44:bb    host-port 1/0/2                Dynamic
100     a8:f9:4b:aa:44:bb    host-port 1/0/2                Dynamic
3 valid mac entries
```

15.НАСТРОЙКА VRF

description

Данная команда используется для назначения описания конфигурируемому экземпляру VRF.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание экземпляра VRF, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRF

Пример

```
rtt(config-vrf)# description "VRF1"
```

ip source-vrf

Данной командой осуществляется перевод системы копирования по TFTP, SCP, FTP, SFTP в VRF.

Использование отрицательной формы команды (no) системы копирования в стандартный режим.

Синтаксис

```
ip { tftp | ftp | scp | sftp | http } source-vrf <VRF>
no ip { tftp | ftp | scp | sftp | http } source-vrf
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

tftp – копирование по протоколу TFTP;

ftp – копирование по протоколу FTP;

scp – копирование по протоколу SCP;

sftp – копирование по протоколу SFTP;

http – копирование по протоколу HTTP.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip tftp source-vrf vrf1
```

ip vrf

Данной командой в системе создается экземпляр VRF и осуществляется переход в режим настройки параметров экземпляра VRF.

Использование отрицательной формы команды (no) удаляет экземпляр VRF из системы.

Синтаксис

```
[no] ip vrf <VRF>
no ip vrf all
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip vrf VRF1
rtt(config-vrf)#
```

ip vrf forwarding

Данной командой задаётся имя экземпляра VRF, в котором будут использоваться указанные сетевой интерфейс, мост, зона безопасности, сервер динамической авторизации (DAS) или группа правил NAT.

Использование отрицательной формы команды (no) удаляет привязку сетевого интерфейса, моста, зоны безопасности или группы правил NAT к экземпляру VRF.

Синтаксис

```
ip vrf forwarding <VRF>
no ip vrf forwarding
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-VTI

CONFIG-GRE

CONFIG-LT

CONFIG-SUBTUNNEL

CONFIG-ZONE

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DAS-SERVER

CONFIG-SNMP-USER

CONFIG-SNMP-VIEW

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-OPENVPN

CONFIG-IPSEC-VPN

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-PEER

CONFIG-IF-E1

CONFIG-IF-MULTILINK

Пример

```
rtt(config-snat-ruleset)# ip vrf forwarding VRF1
```

show ip vrf

Команда используется для просмотра информации о существующих в системе экземплярах VRF.

Синтаксис

show ip vrf [<VRF>]

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip vrf
Name                              Interfaces
--------------------------------  ----------------------------------------
VRF1                              gi1/0/8
VRF2                              gi1/0/10.22
test                              gi1/0/2
```

16.НАСТРОЙКА IP-АДРЕСАЦИИ

ip address

Данной командой создаётся IP-интерфейс и добавляются IP-адрес и маска подсети для конфигурируемого интерфейса (физического интерфейса, группы агрегации каналов, туннеля или сетевого моста).

Использование отрицательной формы команды (no) удаляет IP-адрес с интерфейса. При удалении последнего адреса IP-интерфейс уничтожается.

При создании IP-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IP-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IPv6-интерфейса, то для IP-интерфейса VLAN резервироваться не будет.

Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды ip internal-usage vlan <VLAN_ID>.

Синтаксис

```
ip address <ADDR/LEN> [ secondary ] [ unit <ID> ]
no ip address { <ADDR/LEN> [ unit <ID> ] | all }
```

Параметры

<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

<ID> – номер юнита, принимает значения [1..4];

secondary – ключ указывает, что настроенный адрес является дополнительным IP-адресом. Если это ключевое слово отсутствует, настроенный адрес является основным IP-адресом. Возможно указать до 7 дополнительных IP-адресов.

all – команда удаляет все IP-адреса на интерфейсе.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

CONFIG-WIREGUARD

Пример

```
rtt(config-if-gi)# ip address 192.168.25.25/24
```

ip redirects

Данной командой включается механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения.

Использование отрицательной формы команды (no) отключает механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения.

Синтаксис

```
[no] ip redirects
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Значение по умолчанию

Отправка ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения включена.

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# no ip redirects
```

ip route source-route

Данной командой на маршрутизаторе включается поддержка обработки ip-пакетов с опцией source routing в заголовке.

Использование отрицательной формы команды (no) поддержка обработки пакетов с опцией source routing отключается.

Синтаксис

```
[no] ip route source-route
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip route source-route
```

ip unnumbered

Данной командой включается режим работы интерфейса с использованием IP-адреса, назначенного на другой интерфейс.

Использование отрицательной формы команды (no) отключает режим ip unnumbered.

Синтаксис

```
ip unnumbered { <IF> | <TUN> }
no ip unnumbere
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – наименования туннелей, задаются в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config-if-e1)# ip unnumbered gigabitethernet 1/0/1
rtt(config-if-e1)#
```

ip unreachables

Данной командой включается отправка ICMP-пакетов о недоступности конечного адреса.

Использование отрицательной формы команды (no) отключает возможность отправки ICMP-пакетов о недоступности конечного адреса.

Синтаксис

```
[no] ip unreachables
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Значение по умолчанию

Отправка ICMP-пакетов о недоступности конечного адреса включена.

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# no ip unreachables
```

show ip interfaces

Команда используется для просмотра информации о существующих в системе IP-интерфейсах.

Синтаксис

```
show ip interfaces [ { <IF> | <TUN> | vrf <VRF> [ ip-address <ADDR> ] } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IP-интерфейсы в указанном VRF;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть принимает значения [0..255]. При указании данного параметра будет отображен IP-интерфейс с указанным IP-адресом;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – наименования туннелей, задаются в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IP-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IP-параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IP-параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip interfaces
```

```
IP address                                          Interface           Admin   Link    Type      Precedence
----------------------------------------------------  ------------------  -----  -----  -------  -----------
155.0.0.60/24                                       br10                Up      Up      static    primary
16.0.0.2/24                                         gi1/0/16            Up      Up      DHCP      --
10.0.0.1/8                                          gi1/0/18            Up      Up      static    primary
180.0.0.1/24                                        gi1/0/18            Up      Up      static    secondary
192.168.1.1/24                                      gi1/0/2             Up      Up      static    primary
25.0.0.2/30                                         gre 25              Up      Up      static    primary
10.1.0.2/24                                         ip4ip4 10           Up      Up      static    primary

esr# show ip interfaces gigabitethernet 1/0/16
IP address                                          Interface           Admin   Link    Type      Precedence
----------------------------------------------------  ------------------  -----  -----  -------  -----------
16.0.0.2/24                                         gi1/0/16            Up      Up      DHCP      --
 DHCP Client settings:
 DHCP Server:          N/A
 Lease time(d:h:m):    000:02:00
 Reboot time:          10 seconds
 Retry time:           300 seconds
 Timeout:              60 seconds
 Select timeout:       0 seconds
 Vendor class ID:      N/A
 Ignore options:
    router

 Latest lease contents:
 Lease time(d:h:m):    000:02:00
 DHCP message type:    DHCPACK
 Renew at:             2015-02-25 12:22:24
 Rebind at:            2015-02-25 13:14:09
 Expires at:           2015-02-25 13:29:09
```

17.НАСТРОЙКА IPV6-АДРЕСАЦИИ

ipv6 address

Данной командой создаётся IPv6-интерфейс и добавляется IPv6-адрес и маска подсети для конфигурируемого интерфейса: физического интерфейса, группы агрегации каналов, туннеля или сетевого моста.

Использование отрицательной формы команды (no) удаляет IPv6-адрес с интерфейса. При удалении последнего адреса IPv6-интерфейс уничтожается.

При создании IPv6-интерфейса система резервирует наибольший незанятый VLAN ID, который будет использоваться внутри системы. Для каждого IPv6-интерфейса на Ethernet-порту резервируется VLAN. Если VLAN уже был зарезервирован для IP-интерфейса, то для IPv6-интерфейса VLAN резервироваться не будет.

Можно зарезервировать VLAN ID для внутреннего использования явно с помощью команды ip internal-usage vlan <VLAN_ID>.

Синтаксис

```
ipv6 address <IPV6-ADDR/LEN> [ unit <ID> ]
no ipv6 address {<IPV6-ADDR/LEN>|all} [ unit <ID> ]
```

Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов (включая IP-адреса) на интерфейс. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPv6-адреса;

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 address fc00::1/120
```

ipv6 enable

Данной командой включается поддержка IPv6 на интерфейсе.

Использование отрицательной формы команды (no) отключает поддержку IPv6 на интерфейсе.

Синтаксис

```
[no] ipv6 enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Поддержка IPv6 отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 enable
```

ipv6 nd managed-config-flag

Данной командой устанавливается managed-config-flag в SLAAC-сообщениях. Использование данного флага определяет получение всех IPv6-настроек от DHCPv6-сервера.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
[no] ipv6 nd managed-config-flag
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd managed-config-flag
```

ipv6 nd ns-interval

Данной командой устанавливается интервала отправки IPv6-сообщений поиска соседа (neighbor solicitation) для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd ns-interval <TIME>
no ipv6 nd ns-interval
```

Параметры

<TIME> – интервал отправки, определяется в миллисекундах [1..172800000].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd ns-interval 2000
```

ipv6 nd other-config-flag

Данной командой устанавливается other-config-flag в SLAAC-сообщениях. Использование данного флага подразумевает предоставление IPv6-префикса по протоколу SLAAC, а остальных параметров по протоколу DHCPv6.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
[no] ipv6 nd other-config-flag
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd other-config-flag
```

ipv6 nd prefix

Данной командой устанавливается IPv6-префикс для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no).

Синтаксис

```
ipv6 nd prefix <IPV6-ADDR/LEN> <VLAID-TIME> <PREFER-TIME> [no-autoconfig] [no-
onlink]
no ipv6 nd prefix
```

Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<VLAID-TIME> – допустимое время жизни анонса, определяется в секундах [1..2147483647];

<PREFER-TIME> – предпочтительное время жизни анонса, определяется в секундах [1..2147483647];

[no-autoconfig] – не использовать Neighbor Discovery для установки всех действительных префиксов на канале из объявлений маршрутизатора (RA), полученных на интерфейсе;

[no-onlink] – конфигурирует указанный префикс как не on-link. Префикс будет объявлен с установленным L-битом.

Значение по умолчанию

Префикс не установлен;

valid time – 0;

preffered time – 0;

no-onlink – не установлен;

no-autoconfig – не установлен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd prefix
```

ipv6 nd ra hop-limit

Данной командой устанавливается параметр hop-limit для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd ra hop-limit <HOP-LIMIT>
no ipv6 nd ra hop-limit
```

Параметры

<HOP-LIMIT> – задается в диапазоне [0..255].

Значение по умолчанию

64

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd ra hop-limit 128
```

ipv6 nd ra lifetime

Данной командой устанавливается параметр lifetime для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd ra lifetime <TIME>
no ipv6 nd ra lifetime
```

Параметры

<TIME> – время жизни рассылаемых объявлений маршрутизатора, определяется в секундах [0..9000].

Значение по умолчанию

Не ограничено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd ra lifetime 120
```

ipv6 nd ra max-interval

Данной командой устанавливается максимальный интервал для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd ra max-interval <TIME>
no ipv6 nd ra max-interval
```

Параметры

<TIME> – максимальный интервал рассылки SLAAC-сообщений, определяется в секундах [4..1800].

Значение по умолчанию

600

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd ra max-interval 1200
```

ipv6 nd ra min-interval

Данной командой устанавливается минимальный интервал для рассылаемых SLAAC-сообщений для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd ra min-interval <TIME>
no ipv6 nd ra min-interval
```

Параметры

<TIME> – минимальный интервал рассылки SLAAC-сообщений, определяется в секундах [3..1350].

Значение по умолчанию

200

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd ra min-interval 30
```

ipv6 nd router-preference

Данной командой устанавливается приоритетность маршрутизатора в процессе выбора маршрута по умолчанию для широковещательного домена.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ipv6 nd router-preference <ACTION>
no ipv6 nd router-preference
```

Параметры

<ACTION> – принимает значения:

- low;
- medium;
- high.

Значение по умолчанию

medium

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-LOOPBACK

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config)# ipv6 nd router-preference low
```

ipv6 redirects

Данной командой включается механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IPv6-сети для конкретного IPv6-назначения.

Использование отрицательной формы команды (no) отключает механизм отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IPv6-сети для конкретного IPv6-назначения.

Синтаксис

```
[no] ipv6 redirects
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Значение по умолчанию

Отправки ICMP-сообщений о существовании более приоритетного маршрутизатора в данной IP-сети для конкретного IP-назначения включена.

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# no ipv6 redirects
```

ipv6 route source-route

Данной командой на маршрутизаторе включается поддержка обработки ipv6-пакетов с расширенным заголовком routing.

Использование отрицательной формы команды (no) отключает поддержку обработки ipv6-пакетов с расширенным заголовком routing.

Синтаксис

```
[no] ipv6 route source-route
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 route source-route
```

ipv6 unreachables

Данной командой включается отправка IPv6 ICMP-пакетов о недоступности конечного адреса.

Использование отрицательной формы команды (no) отключает возможность отправки IPv6 ICMP-пакетов о недоступности конечного адреса.

Синтаксис

```
[no] ipv6 unreachables
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отправка IPv6 ICMP-пакетов о недоступности конечного адреса включена.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# no ipv6 unreachables
```

show ipv6 interfaces

Команда используется для просмотра информации о существующих в системе IPv6-интерфейсах.

Синтаксис

```
show ipv6 interfaces { { <IF> | <TUN> | ipv6-address <IPV6-ADDR> } | vrf <VRF> [
ipv6-address <IPV6-ADDR> ] }
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа. При указании данного параметра будут отображены IPv6-интерфейсы в указанном VRF;

<IF> – наименования системных интерфейсов, задаются в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – наименования туннелей, задаются в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128]. При указании данного параметра будут отображен IPv6-интерфейс с указанным IPv6-адресом.

В команде можно указать несколько системных интерфейсов. Если не указывать индексы интерфейсов, то будут отображены все IPv6-интерфейсы, относящиеся к системным интерфейсам указанного типа.

Если в команде указан определенный системный интерфейс, получающий IPv6-параметры по протоколу DHCP, то будут отображены настройки DHCP-клиента и состояние текущей аренды IPv6-параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 interfaces
 IPv6 address                                          Interface             Admin   Link    Type
 ----------------------------------------------------  --------------------  -----   -----   -------
 fc00::2/120                                           gi1/0/5               Up      Up      DHCP
 fe80::aaf9:4bff:feaa:45/64                            gi1/0/5               Up      Up      static

 esr# show ip interfaces gigabitethernet 1/0/16
 IPv6 address                                          Interface             Admin   Link    Type
 ----------------------------------------------------  --------------------  -----   -----   -------
 fc00::2/120                                           gi1/0/16              Up      Up      DHCP
  DHCP Client settings:
  DHCP Server:          N/A
  Lease time(d:h:m):    000:02:00
  Reboot time:          10 seconds
  Retry time:           300 seconds
  Timeout:              60 seconds
  Select timeout:       0 seconds
  Vendor class ID:      N/A
  Ignore options:
     router
  Latest lease contents:
  Lease time(d:h:m):    000:02:00
  DHCP message type:    DHCPACK
  Renew at:             2015-02-25 12:22:24
  Rebind at:            2015-02-25 13:14:09
  Expires at:           2015-02-25 13:29:09
```

18. УПРАВЛЕНИЕ ПРОФИЛЯМИ

address-port pair

Команда используется для задания связки IP-адресов и TCP/UDP-порта.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] address-port pair { <IPv4-ADDR> | <IPv6-ADDR> } : <PORT>
```

Параметры

<IPv4-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – номер порта, принимает значение [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-ADDRESS-PORT

Пример

```
rtt(config-object-group-address-port)# address-port pair 192.168.1.1:23
```

application

Данная команда используется для указания приложений, попадающих под действие данного профиля.

Использование отрицательной формы команды (no) удаляет приложение из текущего профиля.

Синтаксис

```
[no] application < APPLICATION >
```

Параметры

< APPLICATION > – указывает приложение, подпадающее под действие данного профиля, может принимать значения:

- afp - Apple Filing Protocol;
- amazon - Amazon Data Services;
- amazon-video - Amazon Video Services;
- amqp - Advanced Message Queuing Protocol;
- apple - Apple Inc;.
- apple-icloud - Apple iCloud;
- apple-itunes - Apple iTunes;
- apple-push - Apple PUSH notifications;
- apple-store - Apple Store;
- ajp - Apache JServ Protocol;
- signal - Signal Protocol;
- bgp - Border Gateway Protocol;
- bittorrent - BitTorrent;
- bjnp - Canon BJNP protocol;
- cisco-skinny - Cisco Skinny;
- cisco-vpn - Cisco VPN;
- citrix - Citrix;
- cloudflare - Cloudflare Inc;.
- coap - Constrained Application Protocol;
- collectd - Collectd;
- corba - Common Object Request Broker Architecture;
- dce-rpc - Distributed Computing Environment ;/ Remote Procedure Calls
- deezer - Deezer ;(music streaming service).
- dhcp - Dynamic Host Configuration Protocol;
- dhcpv6 - IPv6 Dynamic Host Configuration Protocol;
- dns - Domain Name System;
- dnscrypt - DNSCrypt;
- drda - Distributed Relational Database Architecture;
- dropbox - Dropbox;
- ebay - eBay;
- edonkey - eDonkey;
- egp - Exterior Gateway Protocol;
- facebook - Facebook;
- facebook-messenger - Facebook Messenger;
- ftp-control - File Transfer Protocol ;(control connections)
- ftp-data - File Transfer Protocol ;(data connections)
- git - Git ;(version control system)
- github - GitHub;, Inc.
- gmail - Gmail;
- gnutella - Gnutella P2P;
- google - Google Inc;.
- google-maps - Google Maps;
- google-meet - Google Meet;
- google-drive - Google Drive;
- google-docs - Google Docs;
- google-services - Google Services;
- gre - Generic Routing Encapsulation;
- gtp - GPRS Tunneling Protocol;
- h323 - H323 Protocol;
- half-life2 - Half-Life 2;

- imo - IMO Messenger;
- hotspotshield - Hotspot Shield;
- http - HTTP;
- http-connect - HTTP tunnel;
- http-proxy - HTTP proxy;
- https - HTTPS;
- iax - IAX;
- icecast - Icecast;
- icmp - ICMP;
- icmpv6 - ICMP IPv6;
- iflix - iflix services;
- igmp - IGMP;
- imap - IMAP;
- imaps - IMAP over TLS;/SSL
- instagram - Instagram;
- ip-in-ip - IP in IP;
- ipp - Internet Printing Protocol;
- ipsec - IPsec;
- irc - Internet Relay Chat;
- jabber - Jabber;
- kakaotalk - KakaoTalk;
- kakaotalk-voice - KakaoTalk voice;
- kaspersky-lab - Kaspersky Laboratory;
- kaspersky-update - Kaspersky Anti-Virus Update;
- kerberos - Kerberos;
- lastfm - LastFM;
- ldap - Lightweight Directory Access Protocol;
- llmnr - Link-local Multicast Name Resolution;
- linkedin - Linkedin;
- lotusnotes - Lotus Notes;
- mdns - Multicast DNS;
- megaco - Megaco;
- mgcp - Media Gateway Control Protocol;
- microsoft - Microsoft services;
- microsoft365 - Microsoft365;
- mining - Mining protocols;
- mpeg-ts - MPEG-TS;
- mqtt - Message Queue Telemetry Transport;
- ms-netlogon - Microsoft Netlogon;
- ms-onedrive - Microsoft OneDrive;
- ms-outlook - Microsoft Outlook;
- ms-sms - Microsoft System Management Server;
- mssql-tds - Tabular Data Stream;
- mysql - MySQL;
- netbios - NetBIOS;
- netflix - Netflix;
- netflow - Netflow;
- nest-log-sink - Nest Log Sink Protocol;
- nfs - Network File System;
- noe - New Office Environment protocol;

- ntp - Network Time Protocol;
- ookla - Ookla services;
- opendns - OpenDNS;
- openvpn - OpenVPN;
- oracle - Oracle Database;
- ospf - Open Shortest Path First;
- pandora - Pandora Radio;
- playstore - Google Playstore;
- pop - Post Office Protocol;
- pops - Post Office Protocol over TLS;/SSL
- postgresql - PostgreSQL;
- pptp - Point-to-Point Tunneling Protocol;
- quic - QUIC;
- radius - Remote Authentication in Dial-In User Service;
- rdp - Remote Desktop Protocol;
- resp - Redis Serialization Protocol;
- rsync - Remote Synchronization;
- rtcp - Real-Time Transport Control Protocol;
- rtmp - Real Time Messaging Protocol;
- rtp - Real-time Transport Protocol;
- rtp-audio - Real-time Transport Protocol audio content;
- rtp-dynamic - Real-time Transport Protocol dynamic content;
- rtp-video - Real-time Transport Protocol video content;
- rtsp - Real time streaming protocol;
- sctp - Stream Control Transmission Protocol;
- sflow - sFlow;
- sip - Session Initiation Protocol;
- skype-teams - Skype and Microsoft Teams;
- smb - Server Message Block;
- smbv23 - Server Message Block Version 23;
- smpp - Short Message Peer-to-Peer;
- smtp - Simple Mail Transfer Protocol;
- smtps - Simple Mail Transfer Protocol over TLS;/SSL
- snmp - Simple Network Management Protocol;
- socks - SOCKS Protocol;
- soundcloud - Soundcloud;
- ssdp - Simple Service Discovery Protocol;
- ssh - Secured Shell;
- starcraft - StarCraft;
- steam - Steam;
- stun - Session Traversal Utilities for NAT;
- syslog - System Logging Utility;
- teamspeak - TeamSpeak;
- teamviewer - TeamViewer;
- telegram - Telegram;
- telnet - Teletype network;
- teredo - Teredo tunneling;
- tftp - Trivial File Transfer Protocol;
- tor - The Onion Router;
- tls - Transport Layer Security Protocol;

- twitch - Twitch;
- twitter - Twitter;
- viber - Viber;
- vmware-vsphere - VMWare vSphere;
- vnc - Virtual Network Computing;
- vrrp - Virtual Router Redundancy Protocol;
- warcraft3 - Warcraft 3;
- wechat - WeChat;
- whatsapp - WhatsApp;
- whatsapp-call - WhatsApp call;
- whatsapp-files - WhatsApp file;
- whois-das - Whois;/DAS
- wikipedia - Wikipedia;
- windows-update - Windows update services;
- world-of-warcraft - World of Warcraft;
- xbox - Xbox services;
- xdmcp - X Display Manager control protocol;)
- yahoo - Yahoo services;
- youtube - YouTube;
- zabbix - Zabbix;
- zeromq - ZeroMQ;
- zoom - Zoom call.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-APPLICATION

Пример

```
rtt(config-object-group-application)# application egp
```

category

Команда используется для задания категорий контентной фильтрации.

Использование отрицательной формы команды (no) удаляет заданную категорию.

Синтаксис

```
[no] category <CATEGORY>
```

Параметры

Мастер-категория включает в себя все свои подкатегории.

<CATEGORY> – имя категории. Принимает значения:

- addictive-substances – мастер-категория, включающая контент, связанный с алкоголем, табаком, электронными сигаретами и психотропными веществами.
    - addictive-substances alcohol – контент, связанный с алкогольной продукцией, производством, потреблением, рекламой, продажей, сайтами производителей.
    - addictive-substances narcotics – контент, связанный с наркотическими, психотропными и одурманивающими веществами, инструкцией по применению, действием, сопровождением немедицинской зависимости.
    - addictive-substances tobacco – контент, связанный с табачными изделиями, электронными сигаретами, описанием использования, рекламой, продажей.
- adware – контент, связанный с рекламным или потенциально нежелательным программным обеспечением (PUA).
- blocked-by-legistations – мастер-категория, включающая контент, заблокированный законодательством отдельных стран.
    - blocked-by-legistations bgc-list – контент, заблокированный Бельгийской комиссией по азартным играм.
    - blocked-by-legistations japanese-police – контент, запрещенный Японской полицией.
    - blocked-by-legistations sia-japan – пиратские сайты, запрещенные Ассоциацией безопасного Интернета (SIA) Японии.
- downloadable-content – мастер-категория, включающая контент связанный с загрузкой пакетов программного обеспечения.
    - downloadable-content file-sharing – контент, связанный с хранением, синхронизацией и обменом файлами пользователя.
    - downloadable-content media-content – контент, связанный с распространением аудио и видео материалов, фильмов, спортивных трансляций, концертов, песен, клипов, обучающих программ.
    - downloadable-content torrents – контент, связанный с Торрент-ресурсами для одноранговового обмена файлами.
- dyn-dns – служба динамического DNS.
- eating-disorders – контент, связанный с расстройствами пищевого поведения, пропагандой или одобрением такого поведения.
- education – мастер-категория, включающая контент связанный с веб-сайтами с образованием.
    - education books – контент, связанный с писателями, книгами, дискуссиями, фан-клубами, издателями, онлайн-библиотеками, книжными магазинами.
    - education educational-institutions – контент, связанный с школами, университетами, образовательными организациями, онлайн-классами, образовательными веб-сервисами.
    - education encyclopedias – контент, связанный с сервисами онлайн-обучения, образовательными порталами, энциклопедиями.
- gambling – мастер-категория, включающая контент связанный с азартными играми, лотереями, информацией, поощряющей участие в этих мероприятиях.
    - gambling casino – контент, связанный с казино, азартными карточными играми, инструкцией по игре, советами, информацией об оборудовании казино.
    - gambling online-betting – контент, связанный с букмекерскими конторами, ставками на спортивные события, скачками, другими событиями.
    - gambling online-lotteries – контент, связанный с онлайн лотереями, информацией о лотереях.
- hate-speech – мастер-категория, включающая контент связанный с ненавистью, дискриминацией, расизмом, экстремизмом, членовредительством, самоубийством.

  - hate-speech extremism – контент, связанный с экстремизмом, терроризмом, расизмом, дискриминацией, ненавистью.
  - hate-speech profane-language – контент, связанный с нецензурной речью.
  - hate-speech self-harm – контент, связанный с угрозами жизни, членовредительствами или самоубийством.
  - hate-speech violence – контент, связанный с оскорблениями, насилием, жестокостью и тревожным контентом с участием людей, животных и экстремистских организаций.
- hobbies-recreation – мастер-категория, включающая контент связанный с хобби, отдыхом, путешествиями, музыкой, домашними животными, развлечениями, кино, театром, телевидением, шоу-бизнесом, фан-клубами, новостями о знаменитостях, сплетнями, комиксами.
  - hobbies-recreation animals – контент, связанный с животными, сообществами, организациями, инструкцией по уходу за животными, онлайн-трансляции.
  - hobbies-recreation cultural-heritage – контент, связанный с культурным наследием, изобразительным искусством, исполнительским искусством, музеями, галереями, выставками.
  - hobbies-recreation games – контент, связанный с информация о видеоиграх, обсуждениях, онлайн-играх, браузерных играх, поддержке игрового программного обеспечения.
  - hobbies-recreation humour – контент, связанный с юмором, анекдотами, мемами, юмористическими историями, смешными видео, комедийными шоу.
  - hobbies-recreation hunting-fishing – контент, связанный с охотой и рыбалкой, сообществами, советами, магазинами снаряжения.
  - hobbies-recreation motor-vehicles – контент, связанный с информацией об автомобилях, обзорах, покупках, модификациях, ремонте, общественном транспорте.
  - hobbies-recreation music – контент, связанный с музыкой, текстами песен, нотами, официальными страницами музыкантов, билетами на концерты, музыкальными заведениями, стриминговыми платформами, музыкальными теле/радиоканалы.
  - hobbies-recreation traveling – контент, связанный с путешествиями, гидами, бронированием отелей и билетов, обзорами, туристическими агентствами, обсуждениями туризма.
  - hobbies-recreation tv-radio – контент, связанный с телекомпаниями, онлайн ТВ, радиокомпаниями, онлайн радиовещателями.
- human-life – мастер-категория, включающая контент связанный с религией, политикой, законами, домом и семьей, средствами массовой информации, армией, оружием, поиском работы, едой, астрологией и паранормальными явлениями.
  - human-life business – контент, связанный с предприятиями, отраслевыми группами и корпоративными услугами.
  - human-life family – контент, связанный с семейным образом жизни, ведением домашнего хозяйства, проектами DIY, ремонтом дома, украшением дома.
  - human-life finance – контент, связанный с финансами, экономикой, сопутствующими услугами.
  - human-life food – контент, связанный с ресторанами, кафе, рецептами, технологиями приготовления еды, кулинарными сообществами, продуктовыми магазинами.
  - human-life government – контент, связанный с правоохранительными органами, государственными учреждениями, политическими объединениями, адвокатскими конторами, юридическими консультациями.
  - human-life magic – контент, связанный с мистикой и эзотерическими темами, гороскопами, гаданиями, магией, НЛО, паранормальными явлениями.
  - human-life military – контент, связанный с военными, вооруженными силами, подрядчиками, доктринами, дискуссиями по вооруженным конфликтам.

  - human-life news – контент, связанный с публичным новостным контентом, официальными СМИ, информационными службами, агрегаторами новостей, новостными сайтами, создаваемыми пользователями.
  - human-life recruitment – контент, связанный с объединениями работодателей и соискателями работы.
  - human-life religion – контент, связанный с религиозными движениями, организациями, культами и межконфессиональным сотрудничеством.
  - human-life weapons – контент, связанный с оружием, охотой, выживанием, стрелковым спортом, военной тематикой.
- it-services – мастер-категория, включающая контент связанный с ИТ-услугами и технологическим оборудованием, анонимайзерами, поисковыми системами, хостинг, домены, рекламные сети, интернет-сервисы, компьютеры, электроника, информационная безопасность, спам-сайты.
  - it-services ads – контент, связанный с рекламными агентствами, рекламными сетями.
  - it-services anonymizers – контент, связанный с анонимным доступом и обходом ограничений, в том числе VPN.
  - it-services electronics – контент, связанный с информацией о технической и бытовой электронике, новостями, инструкциями, обзорами, сайтами производителей.
  - it-services hosting – контент, связанный с хостингом, регистрацией доменов, услугами поиска WHOIS.
  - it-services internet-services – контент, связанный с поставщиками интернет-инфраструктуры и услуг, Интернет-провайдеров, онлайн-переводчиками, аналитикой веб-сайтов, хостингом для разработки программного обеспечения, веб-сервисами.
  - it-services it-security – контент, связанный с ИТ-безопасностью и продуктами/услугами по обеспечению безопасности.
  - it-services searchers – контент, связанный с базой данных интернет-ресурсов со встроенной поисковой системой.
  - it-services spam – контент, связанный с рассылкой спама или использованием в спаме.
- kids-internet – контент, связанный с сайтами, ориентированными на детей, посвященные образовательным, развлекательным и коммуникационным платформам.
- lifestyle – мастер-категория, включающая контент связанный со спортом, красотой и здоровьем.
  - lifestyle fashion – контент, связанный с модой и стилем, модельные агентства, показы мод, гиды по одежде и обуви.
  - lifestyle health – контент, связанный со здоровым образом жизни, фитнесом, здоровым питанием, альтернативными методами лечения.
  - lifestyle pharmacy – контент, связанный с медициной, аптеками, медицинскими организациями, реабилитационными центрами, фармацевтическими компаниями, лекарствами, БАД, профилактикой инфекций, медицинской психологической помощью.
  - lifestyle sport – контент, связанный со спортом, историей спорта, правилами игры, сообществами, командами, легкой атлетикой, болельщиками, новостями спорта.
- malware – контент, связанный с вредоносными ресурсами и URL-адресами загрузки программного обеспечения.
- payments – мастер-категория, включающая контент связанный с Интернет-магазинами, платежными системами, банками, онлайн-кошельками и криптовалютой.
  - payments cryptocurrency – контент, связанный с покупкой, продажей, хранением и изучением криптовалют и криптомайнинга.
  - payments online-banks – контент, связанный с услугами онлайн-банкинга и электронного кошелька.
  - payments shops – контент, связанный с Интернет-магазинами и аукционами по продаже товаров или услуг.

    - payments transactions – контент, связанный с системами электронных денег, доступом к личным кабинетам и возможность проведения платежей.
- rat-node – контент, связанный с программным обеспечением инструментов удаленного администрирования.
- rental-services – контент, связанный с арендой, продажей и покупкой недвижимости.
- rf-law-prohibited – мастер-категория, включающая контент, заблокированный законодательством Российской Федерации.
    - rf-law-prohibited 436-fz – категория соблюдения законодательства, веб-ресурсы, предусмотренные статьей 5 Федерального закона 436-ФЗ.
    - rf-law-prohibited extremist-rf-minjust – контент, внесенный в федеральный реестр экстремистских материалов, который ведется Министерством юстиции Российской Федерации.
    - rf-law-prohibited roskomnadzor – контент, включенный в единый реестр запрещенных Интернет-ресурсов Роскомнадзора.
- scam – контент, связанный с угрозами потери данных, фишинговыми атаками.
- sex – мастер-категория, включающая контент, связанный с порнографией, эротикой, сексуальностью, нудизмом, половым воспитанием, сексуальными услугами онлайн или оффлайн, продуктами и услугами сексуального характера, информацией об абортах.
    - sex erotic-18+ – контент откровенно сексуального характера, предназначенный для взрослой аудитории, часто представленный в возбуждающей манере.
    - sex lgbtq+ – контент, связанный с LGBTQ+ темами.
    - sex lingerie – контент, связанный с производителями, продавцами нижнего белья и модными страницами с описаниями или фотографиями нижнего белья.
    - sex near-sexual-content – контент, связанный с информацией о сексуальности, сексуальном здоровье, контрацепции и смежных темах, подходящей для взрослой аудитории, но не предназначенной для детей.
- social-media – мастер-категория, включающая контент, связанный с отправкой личных сообщений или публикаций общедоступных сообщений или сообщений для ограниченной аудитории.
    - social-media blogs – контент, связанный с блог-платформами, персональными блогами, сервисами для создания и ведения блогов.
    - social-media chats – контент, связанный с публичными обсуждениями и обменом мгновенными сообщениями в режиме реального времени.
    - social-media dating – контент, связанный с услугами онлайн и оффлайн знакомств.
    - social-media email – контент, связанный со страницами входа в почтовые сервисы и почтовые ящики.
    - social-media social-networks – контент, связанный с созданием, отображением и управлением контактами между людьми, организациями и правительствами.
- tor – контент, связанный с узлами Top.
- uncategorized – некатегоризированный контент.
- vulnerables – контент, связанный с уязвимым программным обеспечением, которым могут воспользоваться злоумышленники.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-CF-KASPERSKY

Пример

```
rtt(config-object-group-cf-kaspersky)# category social-media chats
```

description

Команда используется для изменения описания профиля.

Использование отрицательной формы команды (no) удаляет описание профиля.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

CONFIG-OBJECT-GROUP-SERVICE

CONFIG-OBJECT-GROUP-MAC

CONFIG-OBJECT-GROUP-APPLICATION

CONFIG-OBJECT-GROUP-URL

CONFIG-OBJECT-GROUP-ADDRESS-PORT

CONFIG-OBJECT-GROUP-CONTENT-FILTER

CONFIG-OBJECT-GROUP-MAIL

CONFIG-OBJECT-GROUP-DOMAIN-NAME

Пример

Установить описание для профиля IP-адресов:

```
rtt(config-object-group-network)# description "Internal addresses"
```

domain

Команда используется для задания доменного имени.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] domain <NAME>
```

Параметры

<NAME> – доменное имя, задаётся строкой от 1 до 253 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-DOMAIN-NAME

Пример

```
rtt(config-object-group-domain-name)# domain yandex.ru
rtt(config-object-group-domain-name)# domain ya.ru
```

email

Команда используется для задания почтового домена или адреса почтового ящика.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] email <NAME>
```

Параметры

<NAME> – почтовый домен или адреса почтового ящика, задаётся строкой от 1 до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-MAIL

Пример

```
rtt(config-object-group-email)# email rtt@Rusteletech.ru
rtt(config-object-group-email)# email Rusteletech.ru
```

ip address-range

Команда используется для задания диапазона IP-адресов.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] ip address-range <FROM-ADDR>[-<TO-ADDR>] [ unit <ID> ]
```

Параметры

<FROM-ADDR> – начальный IP-адрес диапазона адресов;

<TO-ADDR> – конечный IP-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IP-адрес.

Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<ID> – номер юнита, принимает значения [1..4];

Возможно указать до 64 отдельных IP-диапазонов в рамках одной группы адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
rtt(config-object-group-network)# ip address-range 192.168.1.1-192.168.1.25
```

ip prefix

Команда используется для задания подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

Синтаксис

```
[no] ip prefix <ADDR/LEN> [ unit <ID> ]
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [8..32];

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
rtt(config-object-group-network)# ip prefix 10.10.10.0/24
```

ipv6 address-range

Команда используется для задания диапазона IPv6-адресов. Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] ipv6 address-range <FROM-ADDR>[-<TO-ADDR>] [ unit <ID> ]
```

Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона адресов;

<TO-ADDR> – конечный IPv6-адрес диапазона адресов, опциональный параметр. Если параметр не указан, то командой задаётся одиночный IPv6-адрес.

Адреса задаются в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<ID> – номер юнита, принимает значения [1..4];

Возможно указать до 64 отдельных IP-диапазонов в рамках одной группы адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
rtt(config-object-group-network)# ipv6 address-range fc00::1:1-fc00:1::32
```

ipv6 prefix

Команда используется для задания IPv6-подсети.

Использование отрицательной формы команды (no) удаляет заданную подсеть.

Синтаксис

```
[no] ipv6 prefix <IPV6-ADDR/LEN> [ unit <ID> ]
```

Параметры

<IPV6-ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-NETWORK

Пример

```
rtt(config-object-group-network)# ipv6 prefix fc00::/126
```

mac address

Данная команда используется для задания диапазона MAC-адресов.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
[no] mac address <MAC-ADDR> <MAC-MASK>
```

Параметры

<MAC-ADDR> – MAC-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<MAC-MASK> – маска MAC-адреса, опциональный параметр, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске. Значение маски по умолчанию FF:FF:FF:FF:FF:FF.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-MAC

Пример

```
rtt(config-object-group-mac)# mac address a8:f9:4b:80:e7:00 FF:FF:FF:FF:FF:00
```

object-group address-port

Команда предназначена для создания профиля связок IP-адресов и TCP/UDP-портов. Профили используются при настройке сервисов, работающих с пулами IP-адресов и TCP/UDP-портами – NAT, Firewall.

Использование отрицательной формы команды (no) удаляет профиль IP-адресов.

Синтаксис

```
[no] object-group address-port <NAME>
```

Параметры

<NAME> – имя конфигурируемого профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра "all" будут удалены все профили IP-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

Создание профиля IP-адресов с именем *remote* и переход в режим конфигурирования профиля:

```
rtt(config)# object-group address-port WEB
```

object-group application

Данная команда используется для создания профиля приложений. Данный профиль используется для фильтрации по приложениям (DPI).

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group application <NAME>
```

Параметры

<NAME> – имя профиля приложений, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили приложений.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group application OGA045
```

object-group content-filter

Данная команда используется для создания профиля категорий контентной фильтрации. Данный профиль используется в наборе пользовательских правил.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group content-filter <NAME>
```

Параметры

<NAME> – имя профиля контентной фильтрации, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили контентной фильтрации.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group content-filter OGC042
```

object-group domain-name

Команда предназначена для создания профиля доменных имен.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group domain-name <NAME>
```

Параметры

<NAME> – наименование профиля доменных имен, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили доменных имен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group domain-name YANDEX
```

object-group mac

Данная команда используется для создания профиля MAC-адресов. Данный профиль используется в MAC-based VLAN.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group mac <NAME>
```

Параметры

<NAME> – имя профиля MAC-адресов, задается строкой до 31 символа. При удалении вместо имени возможно использовать ключ all. При использовании ключа "all" будут удалены все профили MAC-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group mac OGM007
```

object-group network

Команда предназначена для создания профиля IP-адресов. Профили используются при настройке сервисов, работающих с пулами IP-адресов – например, NAT, Firewall, Remote-Access, также для создания списка префиксов.

Использование отрицательной формы команды (no) удаляет профиль IP-адресов.

Синтаксис

```
[no] object-group network <NAME>
```

Параметры

<NAME> – имя конфигурируемого профиля IP-адресов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили IP-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

Создание профиля IP-адресов с именем *remote* и переход в режим конфигурирования профиля:

```
rtt(config)# object-group network remote
```

object-group service

Команда предназначена для создания профиля TCP/UDP-портов. Данный профиль используется в правилах сервисов NAT и Firewall.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group service <NAME>
```

Параметры

<NAME> – наименование профиля портов, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили TCP/UDP-портов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group service ssh
```

object-group url

Команда предназначена для создания профиля URL-ссылок.

Использование отрицательной формы команды (no) удаляет профиль.

Синтаксис

```
[no] object-group url <NAME>
```

Параметры

<NAME> – наименование профиля url, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все профили URL-ссылок.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# object-group url vk
```

port-range

Командой задаётся диапазон TCP/UDP-портов, относящихся к профилю.

Использование отрицательной формы команды (no) удаляет запись из конфигурируемого профиля.

Синтаксис

```
port-range <PORT>
no port-range [<PORT> | all]
```

Параметры

<PORT> – номер порта, принимает значение [1..65535].

Можно указать несколько портов перечислением через запятую «,» либо указать диапазон портов через «-». Пример записи: <PORT>, <PORT> или <PORT>-<PORT> или <PORT>-<PORT>, <PORT>-<PORT>.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-SERVICE

Пример

```
rtt(config-object-group-service)# port-range 22
```

regexp

Данной командой описывается шаблон URL-ссылок.

Использование отрицательной формы команды (no) удаляет шаблон URL-ссылок.

Синтаксис

```
regexp <REGEXP>
no regexp {<REGEXP>|all}
```

Параметры

<REGEXP> – регулярное выражение. Описывается строкой до 255 символов. Символ "\" должен быть экранирован.

all – ключ, используемый для удаления всех созданный правил.

Значение по умолчанию

Шаблон не создан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-URL

Пример

```
rtt(config-object-group-url)# '^http:\/\/site\.ru'
```

show object-group

Данная команда используется для просмотра информации о профилях IP-адресов и TCP/UDP-портов.

Синтаксис

```
show object-group <PROFILE_TYPE> [<NAME>]
```

Параметры

<PROFILE_TYPE> – тип профиля:

- address-port – профиль связок IP-адресов и TCP/UDP-портов;
- application – профиль приложений;
- content-filter – профиль контентной фильтрации;
- domain-name – профиль доменных имен;
- email – профиль почтовых доменов и адресов почтовых ящиков;
- mac – профиль профиля MAC-адресов;
- network – профиль IP-адресов;
- service – профиль TCP/UDP-портов;
- url – профиль URL-ссылок.

<NAME> – имя профиля, задаётся строкой до 31 символа, опциональный параметр. Если имя профиля не задано, то будет выведен список профилей соответствующего типа и их описание.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show object-group network
Network                          Description
-------------------------------  --------------------------------
remote                           --
local                            --
tunnel                           --
rtt# show object-group network remote
IP Addresses
--------------------------------
10.102.0.0/16
rtt# show object-group service
```

```
Service                          Description
-------------------------------  -------------------------------
telnet                           --
ssh                              --
dhcp_server                      --
dhcp_client                      --
ntp                              --
rtt# show object-group service ssh
Port ranges
-------------------------------
22
```

url

Команда используется для задания URL-ссылки.

Использование отрицательной формы команды (no) удаляет ссылку из конфигурируемого профиля.

Синтаксис

```
url <URL>
no url [ <URL> | all ]
```

Параметры

<URL> – текстовое поле, содержащее URL-ссылку длиной от 4 до 255 символов.

При удалении с использованием ключа "all" будут удалены все внесенные ранее URL-ссылки.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-URL

Пример

```
rtt(config-object-group-url)# url https://vk.com
```

vendor

Команда используется для задания поставщика категорий контентной фильтрации.

Использование отрицательной формы команды (no) удаляет категории контентной фильтрации данного поставщика.

Синтаксис

```
[no] vendor <CONTENT-FILTER-VENDOR>
```

Параметры

<CONTENT-FILTER-VENDOR> – название поставщика категорий контентной фильтрации. Принимает значения:

Kaspersky-Lab – в текущей версии ПО в качестве поставщика категорий контентной фильтрации может выступать только Лаборатория Касперского.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OBJECT-GROUP-CONTENT-FILTER

Пример

```
rtt(config-object-group-content-filter)# vendor kaspersky-lab
```

19. УПРАВЛЕНИЕ NAT

action destination-nat

Данной командой выполняется трансляция адреса и порта получателя для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }
no action destination-nat
```

Параметры

off – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

pool <NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP-портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и TCP/UDP-порт получателя на значения, выбранные из пула;

netmap <ADDR/LEN> – IP-подсеть, используемая при трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес получателя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

Пример

```
rtt(config-dnat-rule)# action destination-nat netmap 10.10.10.0/24
```

action source-nat

Данной командой назначается тип действия «трансляция адреса и порта отправителя» и параметры трансляции для трафика, удовлетворяющего критериям, заданным командами «match».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
action source-nat { off | pool <NAME> | netmap <ADDR/LEN> [static] | interface
[FIRST_PORT – LAST_PORT] }
no action source-nat
```

Параметры

off – трансляция отключена. Трафик, попадающий под заданные критерии, не будет изменен;

pool <NAME> – задаёт пул IP-адресов и/или TCP/UDP-портов. У трафика, попадающего под заданные критерии, будет изменен IP-адрес и/или TCP/UDP-порт отправителя на значения, выбранные из пула;

netmap <ADDR/LEN> – задаёт IP-подсеть для трансляции. У трафика, попадающего под заданные критерии, будет изменен IP-адрес отправителя на IP-адрес из указанной подсети. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

static – включение статического NAT, доступно при использовании netmap;

interface [FIRST_PORT – LAST_PORT] – задаёт трансляцию в IP-адрес интерфейса. У трафика, попадающего под заданные критерии, будет изменён IP-адрес отправителя на IP-адрес интерфейса, в который трафик будет отправлен. Если дополнительно задан диапазон TCP/UDP-портов, то трансляция будет происходить ещё и для TCP/UDP-портов отправителя, они будут заменены на указанный диапазон портов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# action source-nat netmap 10.10.10.0/24
```

check output-interface

Данной командой разрешается очистка SNAT-сессий, в случае если выходной интерфейс изменен.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
check output-interface
no check output-interface
```

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# check output-interface
```

description

Данной командой задаётся описание.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

CONFIG-DNAT-POOL

CONFIG-SNAT-POOL

Пример

```
rtt(config-snat-ruleset)# description "test ruleset"
```

enable

Данной командой активируется конфигурируемое правило.

Отрицательная форма команды (no) деактивирует использование правила.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# enable
```

from

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего из определенной зоны или интерфейса.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

Синтаксис

```
from { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no from
```

Параметры

<NAME> – имя зоны изоляции, задается строкой до 12 символов.

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.

Группа правил со значением «default» параметра «from» может быть только одна.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

Пример

```
rtt(config-dnat-ruleset)# from zone untrusted
```

ip address

Данной командой устанавливается внутренний IP-адрес, на который будет заменяться IP-адрес получателя.

Использование отрицательной формы команды (no) удаляет заданный IP-адрес.

Синтаксис

```
ip address <ADDR>
no ip address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-POOL

Пример

```
rtt(config-dnat-pool)# ip address 10.10.10.10
```

ip address-range

Данной командой задаётся диапазон внешних IP-адресов, на которые будет заменяться IP-адрес отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон адресов.

Синтаксис

```
ip address-range <IP>[-<ENDIP>]
no ip address-range
```

Параметры

<IP> – IP-адрес начала диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ENDIP> – IP-адрес конца диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для трансляции используется только IP-адрес начала диапазона.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
rtt(config-snat-pool)# ip address-range 10.10.10.1-10.10.10.20
```

ip nat proxy-arp

Данная команда позволяет маршрутизатору отвечать на ARP-запросы IP-адресов из указанного пула. Функция необходима для того, чтобы не назначать все IP-адреса из пула трансляции на интерфейсе.

Синтаксис

```
ip nat proxy-arp <OBJ-GROUP-NETWORK-NAME>
no ip nat proxy-arp
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Функция NAT Proxy ARP отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-IG

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip nat proxy-arp nat-pool
```

ip port

Данной командой устанавливается внутренний TCP/UDP-порт, на который будет заменяться TCP/UDP-порт получателя.

Использование отрицательной формы команды (no) удаляет заданный TCP/UDP-порт.

Синтаксис

```
ip port <PORT>
no ip port
```

Параметры

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-POOL

Пример

```
rtt(config-dnat-pool)# ip port 5000
```

ip port-range

Данной командой задаётся диапазон внешних TCP/UDP-портов, на которые будет заменяться TCP/UDP-порт отправителя.

Использование отрицательной формы команды (no) удаляет заданный диапазон портов.

Синтаксис

```
ip port-range <PORT>[-<ENDPORT>]
no ip port-range
```

Параметры

<PORT> – TCP/UDP-порт начала диапазона, принимает значения [1..65535];

<ENDPORT> – TCP/UDP-порт конца диапазона, принимает значения [1..65535]. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для трансляции используется только TCP/UDP-порт начала диапазона.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
rtt(config-snat-pool)# ip port-range 20-100
```

match destination-address

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] destination-address { address-range { <ADDR>[-<ADDR>] | <IPV6-ADDR>[-<IPV6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-group network <OBJ-GROUP-NETWORK-NAME> | any }
no match destination-address
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match destination-address object-group remote
```

match destination-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для связок IP-адресов и TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
no match destination-address
```

Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535].

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match destination-address object-group local
```

match destination-port

Данной командой устанавливается профиль TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] destination-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME> | any }
no match destination-port
```

Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match destination-port object-group ssh
```

match icmp

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой «match protocol». Командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
no match icmp
```

Параметры

<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0 ..255];

<ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0 ..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP;

<OPTION> – стандартные типы ICMP-сообщений, может принимать значения:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;
- timestamp-request;
- traceroute.

Значение по умолчанию

any any

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match icmp 2 any
```

match protocol

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] protocol <TYPE>
no match protocol
match [not] protocol-id <ID>
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, icmp6, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match protocol udp
```

match source-address

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] source-address { address-range { <ADDR>[-<ADDR>] | <IPV6-ADDR>[-<IPV6-
ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-group network <OBJ-
GROUP-NETWORK-NAME> | any }
no match source-address
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match source-address object-group local
```

match source-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для связок IP-адресов и TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT>
} | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
no match source-address-port <OBJ-GROUP-ADDRESS-PORT-NAME>
```

Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535].

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match source-address-port object-group admin
```

match source-port

Данной командой устанавливается профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании команды «not» правило будет срабатывать для TCP/UDP-портов отправителя, которые не входят в указанный профиль. Использование отрицательной формы команды (no) отменяет установленное действие.

Синтаксис

```
match [not] source-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME>
| any }
no match source-port
```

Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULE

CONFIG-SNAT-RULE

Пример

```
rtt(config-snat-rule)# match source-port object-group telnet
```

nat alg

Данная команда включает функцию трансляции IP-адресов в заголовках уровня приложений.

Использование отрицательной формы команды (no) отключает функцию трансляции IP-адресов в заголовках уровня приложений.

Синтаксис

```
[no] nat alg { <PROTOCOL> }
```

Параметры

<PROTOCOL> – протокол уровня приложений, в заголовках которого должна работать трансляция адресов, принимает значения [ftp, h323, pptp, netbios-ns, gre, sip, tftp].

Вместо имени отдельного протокола можно использовать ключ "all", который включает трансляцию IP-адресов в заголовках всех доступных протоколов.

Значение по умолчанию

Функцию трансляции IP-адресов в заголовках уровня приложений отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# nat alg ftp
```

nat destination

Данная команда позволяет войти в режим настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов получателя (DNAT, Destination NAT).

Синтаксис

```
[no] nat destination
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# nat destination
rtt(config-dnat)#
```

nat source

Данная команда позволяет войти в режим настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

Использование отрицательной формы команды (no) удаляет настройки сервиса трансляции адресов отправителя (SNAT, Source NAT).

Синтаксис

```
[no] nat source
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# nat source
rtt(config-snat)#
```

persistent

Командой выполняется включение функции NAT persistent.

NAT persistent позволяет приложениям использовать STUN (session traversal utilities for NAT – утилиты проброса сессий для NAT) для установления соединения с устройствами, находящимися за шлюзом NAT. При этом гарантируется, что запросы от одного и того же внутреннего адреса транслируются в один и тот же внешний адрес.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] persistent
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Функция NAT persistent отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-POOL

Пример

```
rtt(config-snat-pool)# persistent
```

pool

Команда создаёт и назначает пул IP-адресов и TCP/UDP-портов с определённым именем для сервиса NAT и меняет командный режим на SNAT POOL или DNAT POOL.

Если пул используется в какой-либо группе правил, то его удалять нельзя.

Использование отрицательной формы команды (no) удаляет заданный пул NAT-адресов.

Синтаксис

```
[no] pool <NAME>
```

Параметры

<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все пулы IP-адресов и TCP/UDP-портов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT

CONFIG-SNAT

Пример

```
rtt(config-snat)# pool nat
rtt(config-snat-pool)#
```

rearrange

Данная команда меняет шаг между созданными правилами.

Синтаксис

```
rearrange <VALUE>
```

Параметры

<VALUE> – шаг между правилами, принимает значения [1..50].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Пример

```
rtt(config-dnat-ruleset)# rearrange 10
```

renumber rule

Данная команда меняет номер правила.

Синтаксис

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

Параметры

<CUR_ORDER> – текущий номер правила, принимает значения [1..10000];

<NEW_ORDER> – новый номер правила, принимает значения [1..10000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Пример

```
rtt(config-dnat-ruleset)# renumber rule 13 100
```

rule

Данной командой создается правило с определённым номером и устанавливается режим командного интерфейса SNAT RULE или DNAT RULE. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1 .. 10000]. Если использовать команду для удаления, то при указании значения «all» будут удалены все правила.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT-RULESET

CONFIG-SNAT-RULESET

Пример

```
rtt(config-snat-ruleset)# rule 10
rtt(config-snat-rule)#
```

ruleset

Данная команда используется для создания группы правил с определённым именем и перехода в командный режим SNAT RULESET или DNAT RULESET.

Использование отрицательной формы команды (no) удаляет заданную группу правил.

Синтаксис

```
[no] ruleset <NAME>
```

Параметры

<NAME> – имя группы правил, задаётся строкой до 31 символа. Если использовать команду для удаления, то при указании значения «all» будут удалены все группы правил.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DNAT

CONFIG-SNAT

Пример

```
rtt(config-snat)# ruleset wan
rtt(config-snat-ruleset)#
```

show ip nat alg

Данная команда используется для просмотра информации о функционале трансляции IP-адресов в заголовках уровня приложений.

Синтаксис

```
show ip nat alg
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip nat alg
ALG Status:
    FTP:   Enabled
    H.323: Disabled
    GRE:   Disabled
    PPTP:  Disabled
```

```
    SIP:   Disabled
    SNMP:  Disabled
    TFTP:  Disabled
```

show ip nat pool

Данная команда используется для просмотра пулов внутренних и внешних IP-адресов и TCP/UDP-портов.

Синтаксис

```
show ip nat <TYPE> pools
```

Параметры

<TYPE> – тип пулов, для просмотра:

- source – внешние IP-адреса и TCP/UDP-порты;
- destination – внутренние IP-адреса и TCP/UDP-порты.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show nat source pools
   Pools
   ~~~~~
ID     Name                   Ip address          Port      Description   Persi
                                                  range                   stent
----   --------------------   -----------------   -------   -----------   -----
0      outside                25.56.48.11         2000 -    outside-poo   false
                                                  3000       l
```

show ip nat proxy-arp

Данная команда используется для просмотра настроек NAT Proxy ARP.

Синтаксис

```
show ip nat proxy-arp
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show nat proxy-arp
Interface     IP address range
----------    ------------------------------------------
gi1/0/15      115.0.0.15-115.0.0.100
```

show ip nat ruleset

Данной командой выполняется просмотр всех или выбранных групп правил, используемых функцией NAT.

Синтаксис

```
show ip nat <TYPE> ruleset [<NAME>]
```

Параметры

<TYPE> – тип группы правил:

- source – группа правил для трансляции IP-адреса и TCP/UDP-порта отправителя;
- destination – группа правил для трансляции IP-адреса и TCP/UDP-порта получателя.

[NAME] – имя группы правил, опциональный параметр. Если имя не задано – будет выведен список всех групп правил.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip nat source rulesets
   Rulesets
   ~~~~~~~~
ID     Name                              To                  Description
----   -------------------------------   -----------------   ---------------
0      factory                           zone 'untrusted'
1      test                              gigabitethernet      test
                                           1/0/1
rtt# show ip nat source rulesets factory
Ruleset:           factory
Description:
To:                none
Rules:
------
Order:             10
Description:       replace 'source ip' by outgoing interface ip address
```

```
Matching pattern:
    Protocol:      any(0)
    Src-addr:      any
    Dest-addr:     any
Action:          interface port any
Status:            Enabled
------------------------------------------------------------------------------
```

show ip nat translations

Данная команда используется для просмотра сессий трансляции. Для просмотра информации о статистике необходимо включить счетчики (раздел ip firewall mode).

Синтаксис

```
show ip nat translations [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address <ADDR> ] [ outiside-source-address <ADDR> ] [ inside-destination-address <ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ summary ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены сессии трансляций в указанном VRF;

summary – выводит суммарную статистику по сессиям трансляции;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Для Source NAT:

- inside-source-address – ключ для указания IP-адреса источника до трансляции;
- inside-destination-address – ключ для указания IP-адреса назначения на входе в маршрутизатор;
- outiside-source-address – ключ для указания IP-адреса источника после трансляции;
- outside-destination-address – ключ для указания IP-адреса назначения на выходе из маршрутизатора.
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Для Destination NAT:

- inside-source-address – ключ для указания IP-адреса источника на выходе из маршрутизатора;
- inside-destination-address – ключ для указания IP-адреса назначения после трансляции;

- outiside-source-address – ключ для указания IP-адреса источника на входе в маршрутизатор;
- outside-destination-address – ключ для указания IP-адреса назначения до трансляции;
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

Source NAT

```
rtt# show ip nat translations
Prot   Inside source  Inside destination Outside source Outside destination Pkts Bytes
----   ------------  ------------------  -------------  ---------------     -----  -----
icmp   115.0.0.10    1.1.0.2             1.1.0.24       1.1.0.2             3      252
```

Пример 2

Destination NAT

```
rtt#  show ip nat translations
Prot   Inside source  Inside destination Outside source Outside destination Pkts Bytes
----   ------------  ------------------  -------------  ---------------     -----  ----
icmp   1.1.0.2       115.0.0.10          1.1.0.2        1.1.0.16            --     --
```

to

Данной командой ограничивается область применения группы правил. Правила будут применяться только для трафика, идущего в определенную зону или интерфейс.

Использование отрицательной формы команды (no) удаляет ограничение области применения группы правил.

Синтаксис

```
to { zone <NAME> | interface <IF> | tunnel <TUN> | default }
no to
```

Параметры

<NAME> – имя зоны изоляции;

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

default – обозначает группу правил для всего трафика, место назначение которого не попало под критерии других групп правил.

Группа правил со значением «default» параметра «to» может быть только одна.

Значение по умолчанию

None

Необходимый уровень привилегий

10

Командный режим

CONFIG-SNAT-RULESET

Пример

```
rtt(config-snat)# ruleset test
rtt(config-snat-ruleset)# to interface gigabitethernet 1/0/1
```

20. УПРАВЛЕНИЕ ТУНЕЛИРОВАНИЕМ, VPN И УДАЛЕННЫМ ДОСТУПОМ

20.1. Управление ключами и сертификатами

20.1.1. Инфраструктура открытых ключей

challenge-password

Данной командой задаётся пароль, который используется для аутентификации PKI-клиентов на PKI-сервере.

Использование отрицательной формы команды (no) удаляет пароль из конфигурации.

Синтаксис

```
challenge-password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }

no challenge-password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PKI-SERVER

CONFIG-TRUSTPOINT

Пример

```
rtt(config-pki-server)# challenge-password StR0nnGP+ss
```

common-name

Данной командой задается общее имя сертификата.

Использование отрицательной формы команды (no) удаляет общее имя сертификата.

Синтаксис

```
common-name <COMMON-NAME>

no common-name
```

Параметры

<COMMON-NAME> – общее имя, задаётся строкой от 1 до 64 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# common-name ca.company.loc
```

country

Данной командой задается код страны как часть отличительного имени сертификата.

Использование отрицательной формы команды (no) удаляет значение кода страны.

Синтаксис

```
country <COUNTRY>

no country
```

Параметры

<COUNTRY> – код страны, задаётся строкой длиной 2 символа. Рекомендуется использовать двухбуквенные обозначения стран "alpha-2" из стандарта ISO 3166-1.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# country RU
```

crypto pki server

Данной командой осуществляется переход в режим настройки PKI-сервера.

Использование отрицательной формы команды (no) удаляет настройки PKI-сервера из системы.

Синтаксис

```
[no] crypto pki server
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# crypto pki server
```

crypto pki trustpoint

Данной командой добавляется PKI-клиент в систему и осуществляется переход в режим настройки созданного PKI-клиента.

Использование отрицательной формы команды (no) удаляет настройки PKI-клиента из системы.

Синтаксис

```
[no] crypto pki trustpoint <TRUSTPOINT>
```

Параметры

<TRUSTPOINT> – имя PKI-клиента, задаётся строкой от 1 до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# crypto pki trustpoint TP_R1
```

dns

Данной командой задается полное доменное имя хоста в качестве альтернативного имени сертификата.

Использование отрицательной формы команды (no) удаляет доменное имя.

Синтаксис

```
dns <NAME>
```

```
no dns <NAME>
```

Параметры

<NAME> – полное доменное имя хоста (FQDN), задается строкой от 1 до 235 символов. Пример записи доменного имени – router.example.loc.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TRUSTPOINT-SAN

Пример

```
rtt(config-trustpoint-san)# dns r1.company.loc
```

email

Данной командой задается адрес электронной почты в качестве альтернативного имени сертификата.

Использование отрицательной формы команды (no) удаляет адрес электронной почты.

Синтаксис

```
email <EMAIL>

no email <EMAIL>
```

Параметры

<EMAIL> – адрес электронной почты, задается строкой от 6 до 254 символов. Пример записи электронной почты – router@example.loc.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TRUSTPOINT-SAN

Пример

```
rtt(config-trustpoint-san)# email r1@company.loc
```

enable

Данной командой активируется PKI-сервер и PKI-клиент.

Использование отрицательной формы команды (no) деактивирует PKI-сервер и PKI-клиент.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER

CONFIG-TRUSTPOINT

Пример

```
rtt(config-pki-server)# enable
```

fingerprint

Данной командой задаётся цифровой отпечаток сертификата PKI-сервера, к которому выполняет подключение PKI-клиент. В случае несовпадения настроенного отпечатка сертификата с тем, что предоставит PKI-сервер операция PKI будет прервана.

Использование отрицательной формы команды (no) удаляет отпечаток сертификата из конфигурации.

Синтаксис

```
fingerprint <FINGERPRINT>
```

```
no fingerprint
```

Параметры

<FINGERPRINT> – значение цифровой отпечаток сертификата, полученного при помощи алгоритма MD5. Имеет размер 16 байт и задается в виде HEX-строки длиной 32 символа без разделителей (YYYY...) или 47 символов с двоеточием в роли разделителя (YY:YY:YY...).

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRUSTPOINT

Пример

```
rtt(config-trustpoint)# fingerprint 79:D2:B6:7E:DF:77:2D:C5:27:68:99:10:BA:EC:D2:47
```

ipv4

Данной командой задается IPv4-адрес в качестве альтернативного имени сертификата.

Использование отрицательной формы команды (no) удаляет IPv4-адрес.

Синтаксис

```
ipv4 <ADDR>

no ipv4 <ADDR>
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TRUSTPOINT-SAN

Пример

```
rtt(config-trustpoint-san)# ipv4 10.0.113.2
```

ipv6

Данной командой задается IPv6-адрес в качестве альтернативного имени сертификата.

Использование отрицательной формы команды (no) удаляет IPv6-адрес.

Синтаксис

```
ipv6 <IPV6-ADDR>

no ipv6 <IPV6-ADDR>
```

Параметры

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TRUSTPOINT-SAN

Пример

```
rtt(config-trustpoint-san)# ipv6 2002:a00:7102::
```

lifetime

Данной командой задаётся время жизни выписываемых PKI-клиентам сертификатов в днях.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию для времени жизни выписываемых PKI-клиентам сертификатов.

Синтаксис

```
lifetime <DAYS>
```

```
no lifetime
```

Параметры

<DAYS> – количество дней, принимает значения в диапазоне [1..3650].

Значение по умолчанию

365 дней.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER

Пример

```
rtt(config-pki-server)# lifetime 14
```

locality

Данной командой задается название населенного пункта или его территориальной единицы как часть отличительного имени сертификата.

Использование отрицательной формы команды (no) удаляет название населенного пункта или его территориальной единицы.

Синтаксис

```
locality <LOCATION>
```

```
no locality
```

Параметры

<LOCATION> – название населенного пункта или его территориальной единицы, задаётся строкой от 1 до 128 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# locality Moscow
```

organization

Данной командой задается название организации как часть отличительного имени сертификата.

Использование отрицательной формы команды (no) удаляет название организации.

Синтаксис

```
organization <ORGANIZATION>

no organization
```

Параметры

<ORGANIZATION> – название организации, задаётся строкой от 1 до 64 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# organization Company
```

organization-unit

Данной командой задается название подразделения организации как часть отличительного имени сертификата.

Использование отрицательной формы команды (no) удаляет название подразделения организации.

Синтаксис

```
organization-unit <ORGANIZATION-UNIT>

no organization-unit
```

Параметры

<ORGANIZATION-UNIT> – название подразделения организации, задаётся строкой от 1 до 64 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# organization-unit ITC
```

show crypto pki server

Данной командой производится вывод служебной информации о PKI-сервере.

Синтаксис

```
show crypto pki server [ cert | database ]
```

Параметры

cert – при указании ключевого слова "cert" будет выведен текущий сертификат PKI-сервера;

policy – при указании ключевого слова "database" будет выведен список выписанных PKI-сервером клиентских сертификатов;

Без указания ключевых слов вывод команды будет содержать общую информацию о PKI-сервере.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show crypto pki server
Status:                    Enabled
Lifetime days:             14
Certificate fingerprint:   D9:BF:2D:88:D7:17:75:C0:4C:D8:44:7F:50:11:28:07
Source:                    --
Last issued serial number: D5:29:C9:01:EE:25:4F:17:9D:04:F9:87:64:92:CD:B7
Challenge password:        Active
rtt# show crypto pki server cert
Version:                                        3
Serial:                                         1
Subject name:
    CN(commonName):                             ca.company.loc
    O(organizationName):                        Company
    L(localityName):                            Moscow
    ST(stateOrProvinceName):                    Moscow
    C(countryName):                             RU
Issuer name:
```

```
    CN(commonName):                                 ca.company.loc
    O(organizationName):                            Company
    L(localityName):                                Moscow
    ST(stateOrProvinceName):                        Moscow
    C(countryName):                                 RU
Validity period:
    Valid after:                                    2025-11-10 03:49:36
    Invalid after:                                  2035-11-08 03:49:36
Signature:
    Algorithm:                                      sha256WithRSAEncryption
    Value:
23:36:B4:FB:5A:1B:C6:CA:60:FE:90:E5:C4:4F:FE:6A:F9:72:

5F:4A:A4:CD:E8:BC:86:D0:D6:65:EE:99:07:2C:CC:53:5B:3A:

30:BF:7A:51:EF:CF:24:28:C7:8A:AE:F5:60:AA:8D:82:86:F4:

05:88:AA:F4:A8:E3:6D:72:8F:2F:85:0F:A6:1F:88:B8:43:75:

3A:DC:BD:C1:2A:2A:5C:56:AE:56:50:FD:8A:D2:DF:B8:7E:9A:

BE:88:99:76:26:8F:32:09:75:C8:9A:76:2B:98:B0:03:2E:D6:

B3:2B:1F:AB:44:EB:DF:6C:90:37:38:10:69:BE:70:93:56:2F:

AE:46:D6:0F:DF:11:1A:76:F7:69:C1:78:33:11:4D:E2:39:34:

7A:D7:E4:25:80:F1:08:23:C9:B7:AF:28:92:DC:1A:1D:EA:32:

DA:30:54:F2:18:92:F6:28:3D:94:29:F5:54:E4:39:CC:0D:49:

82:24:80:DA:FE:7B:2F:E3:AF:9C:AF:DF:19:DE:B5:80:7C:66:

EC:03:A1:42:73:BC:FD:34:9B:1C:DF:68:36:55:23:94:70:A6:

86:7D:2A:4D:56:92:72:8F:8B:5A:D2:0E:D9:13:93:A0:7C:8E:

79:39:52:81:83:9A:45:4F:EE:E2:51:3A:6D:3F:7D:45:64:84:

D9:A9:3B:8C:7F:73:F5:2F:20:E1:0D:01:3C:E6:44:1E:D3:E9:

DB:72:2E:EA:E7:7B:47:BA:B7:B4:DE:69:CB:27:CF:B1:E4:62:

83:C2:00:22:31:85:1E:B2:EC:61:9C:02:78:A6:C7:F6:28:89:

B1:11:11:2A:CC:7A:CB:57:04:86:23:71:E0:23:6D:45:70:EA:

10:E0:E3:07:69:DA:6E:AD:3D:22:FD:2F:1D:84:8E:E1:65:7A:

E4:C6:33:AE:E7:31:B5:B0:0E:4F:D9:89:49:86:BB:C4:6E:AF:

F7:54:5B:E7:B3:12:2A:E7:95:06:9B:75:E9:57:90:4F:0B:FF:

C4:84:85:60:C8:DB:E9:79:A7:69:C9:CD:1B:7A:69:F5:07:5B:

DB:8A:DD:9F:F0:9A:2B:E4:F9:F5:7E:10:77:95:00:82:52:66:

53:85:74:23:8D:BF:83:3D:18:CA:A9:F8:5C:70:47:C2:A7:C6:

04:AB:82:E6:9B:58:BA:CB:32:77:7E:79:04:AB:7C:12:0C:F5:

E8:7F:BD:20:A8:B6:AF:35:5B:70:5F:5B:ED:6A:99:E3:37:FC:

EA:B4:72:42:B6:88:1D:F5:32:4A:F8:ED:E6:DB:5A:36:52:90:
```

```
B6:78:2C:D7:18:19:78:1C:B3:4A:50:71:82:85:57:9E:71:23:
                                                1A:A1:F8:ED:82:4C:EF:87
Public key info:
    Algorithm:                                  RSA
    Key size:                                   4096
    Exponent:                                   65537
    Modulus:
00:A0:52:45:9F:CA:36:F5:DC:45:3D:23:D5:68:71:13:28:AB:

26:A1:48:0D:77:71:C9:55:E8:3E:F7:62:12:F9:0D:5D:35:FC:

83:27:5F:E0:B2:C2:09:A2:B0:3A:E8:A3:04:AC:2C:E5:92:85:

70:B4:1A:7A:6A:13:D0:49:9E:D3:03:A9:3C:D4:98:8C:9C:A7:

0B:57:A2:84:E1:B3:4A:F3:9B:4E:69:58:08:71:62:39:71:64:

9D:8D:83:D9:6B:49:AE:BA:00:E7:69:51:91:0B:F2:1F:58:62:

CF:0A:BF:84:BE:6A:CB:AE:2B:73:71:AC:97:55:AE:B6:7D:77:

D3:82:2F:DE:DC:42:70:DD:D1:0A:BC:43:05:46:23:0F:02:74:

6A:6A:97:C3:7E:41:7E:6B:2E:A8:E4:11:80:8F:98:45:D4:DB:

D0:05:8C:BB:08:A6:92:82:CB:31:E7:E0:FE:98:D3:98:7A:2F:

A1:5F:53:3F:8D:63:77:68:28:8C:7D:F4:6B:56:43:18:9E:CD:

68:82:31:F5:C1:F5:C7:57:86:75:82:7A:72:94:4D:F7:84:4C:

FC:49:B3:FA:06:EF:A6:D9:21:F1:21:1E:F1:B5:B3:6C:72:AF:

D8:DE:EE:BF:79:64:54:8D:A8:C0:AE:02:F2:F8:D2:DC:49:52:

6E:2D:45:CE:19:01:56:72:10:E5:83:05:FE:59:2E:2E:30:F1:

B7:C3:D6:73:4A:A1:18:7F:57:79:26:AF:67:2E:65:4B:47:7D:

40:B3:C9:E1:19:01:AE:48:B3:2D:A8:7B:12:3F:9C:66:7E:4D:

1D:E8:87:2F:F7:C3:A3:E5:13:D8:D0:EE:B2:95:76:9B:21:10:

7A:0D:01:C2:B5:56:77:8F:17:A2:A6:41:78:C7:C2:CD:F0:DE:

DF:06:61:57:0F:EE:E2:5B:C8:EA:0B:C6:0E:8B:4B:E2:54:24:

34:A2:91:9A:51:9B:00:46:68:C0:19:AD:17:DE:C8:C0:34:AC:

76:F8:83:D8:25:E9:FB:80:4A:CA:D3:27:F5:4B:EA:62:89:A2:

04:BF:3C:50:A8:22:51:8A:B4:06:1F:33:39:EE:B3:9C:8C:10:

12:E9:74:6A:7D:85:F9:88:C3:01:C6:3A:1D:A3:48:11:F5:31:

8D:72:87:BB:E8:B8:12:A8:E4:FD:27:52:FA:79:72:17:2C:EA:

CA:67:C3:80:26:86:DB:78:D7:C4:0B:3C:28:6D:8F:E4:F0:94:

71:B5:FE:FB:A5:A2:C8:6A:04:79:F6:D2:AC:15:63:DE:AA:A2:

AC:C4:BE:D8:66:E4:64:44:01:03:88:B7:66:85:59:DC:23:5F:
                                                FB:53:3B:15:26:45:AE:51:5F
X509v3 Subject key identifier:
```

```
    ID:
07:3A:2A:A0:07:C0:86:B3:8F:E6:0D:B4:38:51:28:09:28:13:
                                                61:0F
    Critical:                                   No
X509v3 Authority key identifier:
    ID:
07:3A:2A:A0:07:C0:86:B3:8F:E6:0D:B4:38:51:28:09:28:13:
                                                61:0F
    Critical:                                   No
X509v3 Basic Constraints:
    CA:                                         Yes
    Critical:                                   Yes
X509v3 Key Usage:
    Usage:                                      Digital Signature
                                                Certificate Sign
                                                CRL Sign
    Critical:                                   Yes
rtt# show crypto pki server database
Serial                                          State       Issue date   Expiration
Revocation   Subject name
                                                                         date         date
----------------------------------------------  ---------   ----------   ----------   ----
------   -------------------------------
D5:29:C9:01:EE:25:4F:17:9D:04:F9:87:64:92:CD:B7   Valid       2025-11-10   2025-11-24   --
/CN=r1.company.loc/O=Company/L
                                                            03:53:11     03:53:11
=Moscow/ST=Moscow/C=RU

rtt#
```

show crypto pki trustpoint

Данной командой производится вывод служебной информации о PKI-клиенте.

Синтаксис

```
show crypto pki trustpoint { NAME } [ cert | ca ]
```

Параметры

NAME – имя PKI-клиента;

cert – при указании ключевого слова "cert" будет выведен текущий сертификат PKI-клиента;

ca – при указании ключевого слова "ca" будет выведен текущий сертификат удостоверяющего центра, который обслуживает запросы текущего PKI-клиента.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show crypto pki trustpoint TP_R1
Name:               TP_R1
Enrollment:         SCEP
Subject name:       /CN=r1.company.loc/O=Company/L=Moscow/ST=Moscow/C=RU
Challenge password: Active
Status:             Ready
Renew date:         2025-11-21 08:41:10
rtt# show crypto pki trustpoint TP_R1 cert
Version:                                        3
Serial:
D5:29:C9:01:EE:25:4F:17:9D:04:F9:87:64:92:CD:B7
Subject name:
    CN(commonName):                             r1.company.loc
    O(organizationName):                        Company
    L(localityName):                            Moscow
    ST(stateOrProvinceName):                    Moscow
    C(countryName):                             RU
Issuer name:
    CN(commonName):                             ca.company.loc
    O(organizationName):                        Company
    L(localityName):                            Moscow
    ST(stateOrProvinceName):                    Moscow
    C(countryName):                             RU
Validity period:
    Valid after:                                2025-11-10 03:53:11
    Invalid after:                              2025-11-24 03:53:11
Signature:
    Algorithm:                                  sha256WithRSAEncryption
    Value:
23:A8:F1:04:15:95:A0:17:EB:A3:95:4F:1B:73:A3:31:B2:95:

FA:06:AA:E7:23:D2:B5:9D:17:1C:C7:A1:90:5A:6F:F9:54:B3:

66:74:90:E3:1C:A3:DB:17:F5:CD:1B:D5:A5:11:9C:86:3C:A3:

87:63:04:9A:FE:B7:7B:A7:07:C8:54:9D:AE:78:D1:75:61:6E:

5F:F4:DC:12:1B:68:0F:53:0D:74:C1:ED:38:96:2D:54:80:6D:

D3:97:71:84:81:82:28:BA:3D:75:70:9B:60:F4:5B:F2:3F:74:

DD:18:76:87:43:A5:1D:74:65:87:49:A9:8F:77:78:45:81:16:

4A:E7:CA:95:A9:31:ED:E0:11:C9:AA:1C:63:3F:11:57:9D:39:

FE:AA:CE:4D:14:E6:75:AE:8C:76:8A:D7:57:F4:B0:8F:8B:FD:

D2:A8:96:4D:73:45:8A:32:12:C5:16:42:DA:29:AF:D5:37:BB:

BF:C1:D8:B3:C8:6E:14:BF:8E:D4:35:34:11:43:4E:F2:6D:F4:

91:E7:F4:05:5F:74:C9:3E:75:4A:5A:9E:E2:6C:83:95:CB:17:

00:95:49:CE:92:4E:33:24:F8:C3:F5:3E:4C:1F:8A:3A:F7:CA:

36:77:66:7D:7D:EB:3F:94:B7:AE:A1:25:AF:73:D0:A9:65:7A:

BE:EE:B2:B1:38:1B:8B:7C:C0:F9:FD:CC:85:22:48:6B:B1:5E:

10:CF:3B:88:FF:02:84:2C:41:02:75:42:F9:39:78:1C:81:E9:

45:B7:6C:3A:53:F5:2E:B5:64:F9:79:E2:4A:E1:A3:DF:00:4D:

63:8F:19:2C:7A:E9:DA:52:83:D5:26:C0:C1:83:46:32:F7:DD:
```

```
F1:EE:5B:21:F3:B9:A7:F5:CD:75:91:EB:6F:C6:7E:7B:E6:28:

52:5D:98:44:25:26:79:B0:C1:6A:43:22:AB:46:48:B6:01:6F:

0F:FB:30:43:C2:CB:A2:8D:73:DF:FC:42:06:74:E0:1F:9C:2A:

DF:0A:1B:02:6A:93:6F:16:00:71:D3:23:6A:DB:9E:CB:5C:57:

00:C3:92:68:98:DC:C5:88:D2:F9:53:4C:79:B5:A1:9E:97:82:

89:74:55:E3:48:55:76:DF:35:96:C5:92:45:85:85:67:7F:D3:

52:E5:EB:DA:A1:6E:2F:04:56:F7:A0:4C:DE:3F:B9:69:87:ED:

10:CF:D9:D5:03:C0:FA:BA:37:FF:D2:D2:A5:54:02:F9:22:E1:

C8:88:82:1E:01:BA:9E:FE:5C:DA:EE:82:56:4B:B7:4D:70:90:

E0:D9:06:8B:C1:E5:C5:B4:96:F9:24:53:D0:B6:4F:2D:32:4D:
                                                    9F:B0:5F:68:AD:F1:65:5E
Public key info:
    Algorithm:                                      RSA
    Key size:                                       2048
    Exponent:                                       65537
    Modulus:
00:E3:03:68:99:6C:D8:84:B0:4F:AF:8F:52:5D:D7:23:2F:D4:

A5:B4:78:DE:5D:CF:B1:44:C4:85:D8:45:71:6E:33:EF:8B:C0:

45:3B:4A:99:A1:52:8F:CB:CC:17:03:25:39:A2:DB:28:7F:BC:

9B:DC:5B:61:04:33:B3:32:6D:4B:03:D2:A9:AE:88:D3:CD:BE:

58:40:68:AC:F6:56:EA:C5:D1:0A:75:D1:59:42:36:F8:41:22:

3E:D4:84:91:E2:C3:36:51:0B:9A:90:22:B3:DF:B2:60:9B:67:

B2:95:5E:C4:41:07:1F:9A:60:18:49:2A:64:D1:F8:60:67:29:

3F:31:AF:B1:EE:DC:10:7F:D3:E5:DE:78:A4:83:53:74:C9:80:

67:80:D4:D3:2F:21:FE:40:C6:57:02:A0:98:99:11:02:E0:44:

1D:02:06:E3:87:24:1C:68:A7:3D:AA:5F:05:49:0C:E7:32:9F:

C6:9A:E9:7B:E3:59:91:F2:C5:22:14:F5:67:5B:2E:EF:41:B7:

5E:15:73:83:DF:43:2D:11:A8:15:8D:48:EF:4B:1C:3A:79:33:

49:76:3A:DB:3F:46:2B:49:47:EB:15:78:F8:50:D6:D5:46:4A:

0F:45:32:D2:50:46:95:F6:03:66:77:81:5A:C2:CE:6A:E6:45:
                                                    2B:2C:03:30:93
X509v3 Subject Alternative Name:
    Names:                                          DNS:r1.company.loc
    Critical:                                       No
X509v3 Subject key identifier:
    ID:
18:B2:C2:83:76:A3:0F:B1:39:4C:7F:C0:C1:A4:7F:46:DA:BF:
                                                    B6:02
    Critical:                                       No
X509v3 Basic Constraints:
    CA:                                             No
    Critical:                                       No
```

```
X509v3 Key Usage:
    Usage:                                          Digital Signature
    Critical:                                       Yes
X509v3 Extended Key Usage:
    Usage:                                          TLS Web Client Authentication
    Critical:                                       No
X509v3 Authority key identifier:
    ID:
07:3A:2A:A0:07:C0:86:B3:8F:E6:0D:B4:38:51:28:09:28:13:
                                                    61:0F
    Critical:                                       No
rtt# show crypto pki trustpoint TP_R1 ca
Version:                                            3
Serial:                                             1
Subject name:
    CN(commonName):                                 ca.company.loc
    O(organizationName):                            Company
    L(localityName):                                Moscow
    ST(stateOrProvinceName):                        Moscow
    C(countryName):                                 RU
Issuer name:
    CN(commonName):                                 ca.company.loc
    O(organizationName):                            Company
    L(localityName):                                Moscow
    ST(stateOrProvinceName):                        Moscow
    C(countryName):                                 RU
Validity period:
    Valid after:                                    2025-11-10 03:49:36
    Invalid after:                                  2035-11-08 03:49:36
Signature:
    Algorithm:                                      sha256WithRSAEncryption
    Value:
23:36:B4:FB:5A:1B:C6:CA:60:FE:90:E5:C4:4F:FE:6A:F9:72:

5F:4A:A4:CD:E8:BC:86:D0:D6:65:EE:99:07:2C:CC:53:5B:3A:

30:BF:7A:51:EF:CF:24:28:C7:8A:AE:F5:60:AA:8D:82:86:F4:

05:88:AA:F4:A8:E3:6D:72:8F:2F:85:0F:A6:1F:88:B8:43:75:

3A:DC:BD:C1:2A:2A:5C:56:AE:56:50:FD:8A:D2:DF:B8:7E:9A:

BE:88:99:76:26:8F:32:09:75:C8:9A:76:2B:98:B0:03:2E:D6:

B3:2B:1F:AB:44:EB:DF:6C:90:37:38:10:69:BE:70:93:56:2F:

AE:46:D6:0F:DF:11:1A:76:F7:69:C1:78:33:11:4D:E2:39:34:

7A:D7:E4:25:80:F1:08:23:C9:B7:AF:28:92:DC:1A:1D:EA:32:

DA:30:54:F2:18:92:F6:28:3D:94:29:F5:54:E4:39:CC:0D:49:

82:24:80:DA:FE:7B:2F:E3:AF:9C:AF:DF:19:DE:B5:80:7C:66:

EC:03:A1:42:73:BC:FD:34:9B:1C:DF:68:36:55:23:94:70:A6:

86:7D:2A:4D:56:92:72:8F:8B:5A:D2:0E:D9:13:93:A0:7C:8E:

79:39:52:81:83:9A:45:4F:EE:E2:51:3A:6D:3F:7D:45:64:84:

D9:A9:3B:8C:7F:73:F5:2F:20:E1:0D:01:3C:E6:44:1E:D3:E9:

DB:72:2E:EA:E7:7B:47:BA:B7:B4:DE:69:CB:27:CF:B1:E4:62:

83:C2:00:22:31:85:1E:B2:EC:61:9C:02:78:A6:C7:F6:28:89:
```

```
B1:11:11:2A:CC:7A:CB:57:04:86:23:71:E0:23:6D:45:70:EA:

10:E0:E3:07:69:DA:6E:AD:3D:22:FD:2F:1D:84:8E:E1:65:7A:

E4:C6:33:AE:E7:31:B5:B0:0E:4F:D9:89:49:86:BB:C4:6E:AF:

F7:54:5B:E7:B3:12:2A:E7:95:06:9B:75:E9:57:90:4F:0B:FF:

C4:84:85:60:C8:DB:E9:79:A7:69:C9:CD:1B:7A:69:F5:07:5B:

DB:8A:DD:9F:F0:9A:2B:E4:F9:F5:7E:10:77:95:00:82:52:66:

53:85:74:23:8D:BF:83:3D:18:CA:A9:F8:5C:70:47:C2:A7:C6:

04:AB:82:E6:9B:58:BA:CB:32:77:7E:79:04:AB:7C:12:0C:F5:

E8:7F:BD:20:A8:B6:AF:35:5B:70:5F:5B:ED:6A:99:E3:37:FC:

EA:B4:72:42:B6:88:1D:F5:32:4A:F8:ED:E6:DB:5A:36:52:90:

B6:78:2C:D7:18:19:78:1C:B3:4A:50:71:82:85:57:9E:71:23:
                                                    1A:A1:F8:ED:82:4C:EF:87
Public key info:
    Algorithm:                                      RSA
    Key size:                                       4096
    Exponent:                                       65537
    Modulus:
00:A0:52:45:9F:CA:36:F5:DC:45:3D:23:D5:68:71:13:28:AB:

26:A1:48:0D:77:71:C9:55:E8:3E:F7:62:12:F9:0D:5D:35:FC:

83:27:5F:E0:B2:C2:09:A2:B0:3A:E8:A3:04:AC:2C:E5:92:85:

70:B4:1A:7A:6A:13:D0:49:9E:D3:03:A9:3C:D4:98:8C:9C:A7:

0B:57:A2:84:E1:B3:4A:F3:9B:4E:69:58:08:71:62:39:71:64:

9D:8D:83:D9:6B:49:AE:BA:00:E7:69:51:91:0B:F2:1F:58:62:

CF:0A:BF:84:BE:6A:CB:AE:2B:73:71:AC:97:55:AE:B6:7D:77:

D3:82:2F:DE:DC:42:70:DD:D1:0A:BC:43:05:46:23:0F:02:74:

6A:6A:97:C3:7E:41:7E:6B:2E:A8:E4:11:80:8F:98:45:D4:DB:

D0:05:8C:BB:08:A6:92:82:CB:31:E7:E0:FE:98:D3:98:7A:2F:

A1:5F:53:3F:8D:63:77:68:28:8C:7D:F4:6B:56:43:18:9E:CD:

68:82:31:F5:C1:F5:C7:57:86:75:82:7A:72:94:4D:F7:84:4C:

FC:49:B3:FA:06:EF:A6:D9:21:F1:21:1E:F1:B5:B3:6C:72:AF:

D8:DE:EE:BF:79:64:54:8D:A8:C0:AE:02:F2:F8:D2:DC:49:52:

6E:2D:45:CE:19:01:56:72:10:E5:83:05:FE:59:2E:2E:30:F1:

B7:C3:D6:73:4A:A1:18:7F:57:79:26:AF:67:2E:65:4B:47:7D:

40:B3:C9:E1:19:01:AE:48:B3:2D:A8:7B:12:3F:9C:66:7E:4D:

1D:E8:87:2F:F7:C3:A3:E5:13:D8:D0:EE:B2:95:76:9B:21:10:

7A:0D:01:C2:B5:56:77:8F:17:A2:A6:41:78:C7:C2:CD:F0:DE:
```

```
DF:06:61:57:0F:EE:E2:5B:C8:EA:0B:C6:0E:8B:4B:E2:54:24:

34:A2:91:9A:51:9B:00:46:68:C0:19:AD:17:DE:C8:C0:34:AC:

76:F8:83:D8:25:E9:FB:80:4A:CA:D3:27:F5:4B:EA:62:89:A2:

04:BF:3C:50:A8:22:51:8A:B4:06:1F:33:39:EE:B3:9C:8C:10:

12:E9:74:6A:7D:85:F9:88:C3:01:C6:3A:1D:A3:48:11:F5:31:

8D:72:87:BB:E8:B8:12:A8:E4:FD:27:52:FA:79:72:17:2C:EA:

CA:67:C3:80:26:86:DB:78:D7:C4:0B:3C:28:6D:8F:E4:F0:94:

71:B5:FE:FB:A5:A2:C8:6A:04:79:F6:D2:AC:15:63:DE:AA:A2:

AC:C4:BE:D8:66:E4:64:44:01:03:88:B7:66:85:59:DC:23:5F:
                                                    FB:53:3B:15:26:45:AE:51:5F
X509v3 Subject key identifier:
    ID:
07:3A:2A:A0:07:C0:86:B3:8F:E6:0D:B4:38:51:28:09:28:13:
                                                    61:0F
    Critical:                                       No
X509v3 Authority key identifier:
    ID:
07:3A:2A:A0:07:C0:86:B3:8F:E6:0D:B4:38:51:28:09:28:13:
                                                    61:0F
    Critical:                                       No
X509v3 Basic Constraints:
    CA:                                             Yes
    Critical:                                       Yes
X509v3 Key Usage:
    Usage:                                          Digital Signature
                                                    Certificate Sign
                                                    CRL Sign
    Critical:                                       Yes
rtt#
```

show crypto pki trustpoints

Данной командой производится вывод списка сконфигурированных PKI-клиентов.

Синтаксис

```
show crypto pki trustpoints
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show crypto pki trustpoints
 Name              Enrollment   Subject name                    Status            Next action
 ---------------   ----------   ------------------------------  ---------------   --------------------
 TP_R1             SCEP         /CN=r1.company.loc/O=Company/L  Ready             2025-11-21 08:41:10
                                =Moscow/ST=Moscow/C=RU
rtt#
```

source-address

Данной командой задается IPv4-адрес, который будет прослушиваться PKI-сервером для обработки входящих запросов. В случае если настройка не будет указана, PKI-сервер будет принимать запросы на всех настроенных IP-интерфейсах маршрутизатора.

Использование отрицательной формы команды (no) удаляет IPv4-адрес.

Синтаксис

```
source-address <ADDR>

no source-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PKI-SERVER

Пример

```
rtt(config-pki-server)# source-address 10.0.113.2
```

source-interface

Данной командой устанавливается сетевой интерфейс, который будет прослушиваться PKI-сервером для обработки входящих запросов.

При использовании отрицательной формы команды (no) удаляет прослушиваемый интерфейс.

Синтаксис

```
source-interface <IF>
no source-interface
```

Параметры

<IF> – тип и идентификатор интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PKI-SERVER

Пример

```
rtt(config-pki-server)# source-interface gigabitethernet 1/0/1
```

state

Данной командой задается название штата, области или провинции.

Использование отрицательной формы команды (no) удаляет название штата, области или провинции.

Синтаксис

```
state <ORGANIZATION>
```

```
no state
```

Параметры

<STATE> – название штата, области или провинции, задаётся строкой от 1 до 128 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER-SUBJECT-NAME

CONFIG-TRUSTPOINT-SUBJECT-NAME

Пример

```
rtt(config-pki-server-subject-name)# state Moscow
```

subject-alt-name

Данной командой осуществляется переход в режим настройки альтернативных имен сертификата PKI-клиента.

Использование отрицательной формы команды (no) удаляет все раннее добавленные альтернативные имена сертификата.

Синтаксис

```
[no] subject-alt-name
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-TRUSTPOINT

Пример

```
rtt(config-trustpoint)# subject-alt-name
```

subject-name

Данной командой осуществляется переход в режим настройки отличительного имени сертификата PKI-сервера или PKI-клиента.

Использование отрицательной формы команды (no) удаляет все раннее добавленные отличительные имена сертификата.

Синтаксис

```
[no] subject-name
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PKI-SERVER

CONFIG-TRUSTPOINT

Пример

```
rtt(config-pki-server)# subject-name
```

url

Команда используется для задания URL-ссылки для подключения к PKI-серверу.

Использование отрицательной формы команды (no) удаляет URL-ссылку из конфигурации PKI-клиента.

Синтаксис

```
url <URL>
no url
```

Параметры

<URL> – текстовое поле, содержащее URL-ссылку длиной от 4 до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRUSTPOINT

Пример

```
rtt(config-trustpoint)# url http://10.0.103.1:80/
```

20.1.2. Ручная генерация

crypto generate cert

Данной командой производится генерация сертификата x.509. Для генерации сертификата необходим файл запроса на подписание сертификата, сертификат удостоверяющего центра (CA), а также приватный ключ.

Синтаксис

```
crypto generate cert csr <CSR> ca <CA> private-key <PRIVATE-KEY> [[invalid-after
<TIME> <DAY> <MONTH> <YEAR>][valid-after <TIME> <DAY> <MONTH> <YEAR>]] filename
<NAME>
```

Параметры

<CSR> – имя файла запроса на подписание сертификата;

<CA> – сертификат удостоверяющего центра (CA);

<PRIVATE-KEY> – приватный ключ;

invalid-after – параметр задаёт дату и время, до которых сертификат считается действительным;

valid-after – данный параметр задаёт дату и время, начиная с которых сертификат считается действительным;

<TIME> – время, указанное в формате ЧЧ:ММ:СС;

<DAY> – день месяца в диапазоне от 1 до 31. Если в месяце нет такого числа, то команда не отработает;

<YEAR> – год в диапазоне от 1970 до 2100 года;

<NAME> – имя сертификата x.509.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# crypto generate cert csr req.pem ca ca.cert private-key private.key filename
cert.crt
```

crypto generate csr

Данной командой производится генерация файла запроса на подписание сертификатов. Для генерации запроса необходимо наличие приватного ключа.

Синтаксис

```
crypto generate csr private-key <PRIVATE-KEY> [[alternative-name <ALT-
NAME>][common-name <COMMON-NAME>][country <COUNTRY>][email-address <E-
MAIL>][locality <CITY>][organization <ORG-NAME>][organizational-unit <ORG-UNIT-
NAME>][state <STATE>]] filename <NAME>
```

Параметры

<PRIVATE-KEY> – имя приватного ключа, задаётся строкой до 31 символа;

<ALT-NAME> – альтернативное имя, задаётся строкой от 5 до 255 символов;

<COMMON-NAME> – общее название, задаётся строкой до 61 символа;

<COUNTRY> – название страны, задаётся строкой до 2 символов;

<E-MAIL> – адрес электронной почты, задаётся строкой от 3 до 64 символов;

<CITY> – название населенного пункта, задаётся строкой до 128 символов;

<ORG-NAME> – название субъекта организации, задаётся строкой до 64 символов;

<ORG-UNIT-NAME> – название организационной единицы, задаётся строкой до 64 символов;

<STATE> – название области, провинции или штата, задаётся строкой до 128 символов;

<NAME> – имя файла запроса на подписание сертификатов.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# crypto generate csr private-key erver.key common-name ec-
client.rtt.Rusteletech.loc country RU email-address rtt@Rusteletech.loc locality
Moscow organization Rusteletech organizational-unit RTT state NSK filename csr.pem
```

crypto generate pfx

Данной командой производится генерация контейнера PKCS12. Для создания контейнера необходим сертификат удостоверяющего центра (CA), сертификат x.509, выпущенный этим удостоверяющим центром и приватный ключ этого сертификата.

Синтаксис

```
crypto generate pfx private-key <PRIVATE-KEY> cert <CERT> ca <CA> [password ascii-text <PASSWORD>] filename <NAME>
```

Параметры

<PRIVATE-KEY> – имя приватного ключа;

<CERT> – имя сертификата;

<CA> – имя сертификата удостоверяющего центра (CA);

<PASSWORD> – пароль для PKCS12-контейнера;

<NAME> – имя генерируемого PKCS12-контейнера, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# crypto generate pfx private-key server.key cert server.crt ca ca.crt password ascii-text password filename gen_rtt.p12
```

crypto generate private-key

Данной командой производится генерация приватного ключа.

Синтаксис

```
crypto generate private-key <KEY-TYPE> [<KEY-SIZE>] filename <NAME>
```

Параметры

<KEY-TYPE> – тип генерируемого ключа:

- rsa – ключ RSA;
- x25519 – ключ формата x25519, совместимый с Wireguard.

<KEY-SIZE> – размер ключа в битах. Значение может находится в диапазоне от 1024 до 4096;

<NAME> – имя ключа, задаётся строкой до 31 символа. Возможна перезапись уже существующих файлов.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# crypto generate private-key rsa 1024 filename gen_private1.key
...................+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
.....................................+++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++
```

crypto generate public-key

Данной командой производится генерация публичного ключа. Для генерации публичного ключа необходимо наличие приватного ключа.

Синтаксис

```
crypto generate public-key x25519 private-key <PRIVATE-KEY> filename <NAME>
```

Параметры

<PRIVATE-KEY> – имя приватного ключа;

<NAME> – имя публичного ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# crypto generate public-key x25519 private-key server.key filename pubkey.pem
```

show crypto certificates

Данной командой выполнятся просмотр количества сертификатов и ключей в энергонезависимой памяти устройства, необходимых для построения VPN (cert, crl, dh, private-key, public-key, pfx и ta).

Синтаксис

```
show crypto certificates
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates
Type               Total
------------       -------
cert               9
dh                 2
private-key        5
public-key         0
ta                 1
crl                1
pfx                0
```

show crypto certificates cert

Данной командой выполнятся просмотр содержимого X.509-сертификатов.

Синтаксис

show crypto certificates cert [<NAME>]

Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа cert.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates cert ca.crt
Version:                                    3
Serial:                                     04:7B:45:DD:3F:0B:00:7F:8D:AB:57:D6:1B:C5:A6:5
6:C0:E
8:
                                            49:E0
Subject name:
    C(countryName):                         RU
    ST(stateOrProvinceName):                Siberia
    L(localityName):                        Moscow
    O(organizationName):                    Rusteletech
    OU(organizationalUnitName):             Rusteletech SC
    CN(commonName):                         ca.test.loc
    emailAddress(emailAddress):             ca@test.loc
Issuer name:
    C(countryName):                         RU
    ST(stateOrProvinceName):                Siberia
    L(localityName):                        Moscow
    O(organizationName):                    Rusteletech
    OU(organizationalUnitName):             Rusteletech SC
    CN(commonName):                         ca.test.loc
    emailAddress(emailAddress):             ca@test.loc
Validity period:
    Valid after:                            2024-06-21 07:39:55
    Invalid after:                          2034-06-19 07:39:55
Signature:
    Algorithm:                              sha256WithRSAEncryption
    Value:                          92:37:D5:4F:3E:A3:93:B3:AF:B1:7A:40:08:9F:7C:80:25:0
5:
                                    89:2D:F7:E5:8F:63:B6:ED:4E:A2:F3:26:E6:57:B6:F2:12:0
5:
                                    1C:E5:42:1F:BC:69:99:F4:88:D4:2A:12:26:21:8E:7B:1D:E
6:
                                    1F:D3:95:FF:63:7C:63:7E:D9:2A:98:BA:C2:ED:F9:01:4F:2
A:
                                    A8:4F:7B:59:60:65:79:1A:37:99:72:F1:C4:65:85:DE:1B:B
B:
                                    E8:0E:C5:F1:C8:95:AE:E6:7B:6C:5F:2C:93:50:51:CC:1B:C
0:
                                    37:50:76:77:60:8B:29:CE:A2:FD:ED:1F:6B:51:CF:A4:67:D
C:
                                    6C:07:0B:EF:80:58:00:C9:C5:72:1E:7D:E2:92:9B:27:2B:5
8:
                                    FE:4B:41:2B:0C:1C:86:86:36:77:D7:99:B8:74:7C:F3:9A:E
2:
                                    0F:83:9B:3C:CE:71:68:B9:07:08:CF:36:9B:5C:8E:18:F9:A
C:
                                    AF:CB:DF:90:2A:2F:40:63:AB:DE:37:40:8F:B8:10:1F:86:1
6:
                                    DC:80:73:42:8D:91:BC:C7:96:6B:42:86:20:AB:EC:43:B7:6
4:
                                    2A:BE:57:9E:CE:E4:1F:A4:61:E1:04:0A:F8:65:AC:C3:2D:3
3:
                                    F6:6F:97:D5:C4:11:6C:40:57:AD:92:4A:C2:01:31:3F:9C:9
5:
                                    5E:F0:2A:65:A3:4A:67:EE:69:F5:A5:B7:6D:AA:99:4B:BE:9
1:
                                    68:8A:CA:BA:BC:9A:10:9B:B6:2D:F7:AE:90:C8:88:A7:84:C
4:
                                    B9:60:20:1B:9B:C3:BF:6C:86:8B:58:DF:62:01:15:F6:A9:7
9:
                                    EE:A6:94:CD:8B:49:06:B5:31:C0:AF:75:BB:2B:45:11:20:3
0:
```

```
                                3E:98:DF:71:32:BB:50:16:13:46:E8:0D:18:4A:BB:76:E1:4
6:
                                61:9A:07:5E:5D:21:56:8D:81:A6:FA:7C:C3:EA:A2:28:E4:4
0:
                                75:6C:2D:2A:42:57:66:5C:FE:3C:E3:54:88:B7:26:BC:15:8
A:
                                CC:F0:1D:FF:22:A5:98:AC:2D:3E:6C:99:CF:98:5F:10:C4:0
6:
                                D4:13:81:FF:79:2B:AE:24:81:09:CB:9C:FC:93:C3:46:96:B
6:
                                BA:29:3D:7C:70:32:19:68:63:50:AB:E2:6C:CE:2E:69:65:D
E:
                                1B:B5:3A:B4:5C:9E:32:74:3D:96:5C:87:F1:3F:6C:55:73:F
A:
                                65:30:91:74:E7:5E:B6:11:61:89:A7:60:5F:51:E3:8E:11:4
6:
                                D4:43:34:58:09:99:A3:55:A5:6E:44:56:DB:3C:82:B9:20:E
2:
                                1A:B2:E9:02:51:DB:39:30:54:28:1F:AE:73:C5:AE:7A:55:4
C:
                                C4:BB:05:F1:0B:CA:60:8B
Public key info:
    Algorithm:                  RSA
    Key size:                   4096
    Exponent:                   65537
    Modulus:                    00:A2:CF:F0:E7:7A:19:0F:05:F5:6E:82:72:F0:AA:C6:2C:8
9:
                                01:F6:48:9E:79:13:ED:7D:4B:86:3D:22:FF:0E:9A:A1:7D:0
D:
                                C6:69:85:C9:0E:6D:34:40:28:9F:E0:F0:9B:C5:C4:21:CA:A
7:
                                43:B5:1B:99:37:B1:F4:61:2F:1B:50:6E:8A:00:88:3D:4B:9
D:
                                F2:D0:C0:C9:CB:E3:01:F2:D5:EB:64:C5:13:CB:F9:1F:47:1
2:
                                67:11:F9:F8:85:FB:B4:37:37:DA:80:B6:88:72:FE:9D:55:1
5:
                                31:DE:F5:C7:83:95:37:9E:0A:AD:B3:53:36:B3:1C:73:CE:5
A:
                                9B:BB:C5:63:11:13:8C:55:B9:54:43:FF:7F:57:60:83:7F:6
2:
                                A0:76:CE:86:56:2C:0F:DD:8C:35:6C:BE:24:78:01:D5:77:F
4:
                                79:97:50:C5:D9:F1:BB:82:73:48:79:A8:08:DD:00:80:66:5
D:
                                F5:DD:E3:FF:E1:FE:30:16:1A:59:74:56:12:74:92:B5:9C:2
8:
                                FF:0B:4A:EE:E4:93:A8:B2:BF:5C:45:70:C5:59:61:9A:D0:7
6:
                                F1:F3:83:31:67:77:48:A0:0D:58:3C:85:EA:55:AD:5A:7D:C
4:
                                7F:62:7D:70:BD:E3:8C:17:0D:AE:55:6F:28:FC:C9:6B:93:F
4:
                                92:E3:C2:FA:9F:99:32:DC:B7:E3:1E:F0:DF:6A:57:0F:AF:2
B:
                                11:4C:8F:17:C2:F0:CC:23:AD:A3:27:C6:E9:CC:90:50:21:7
9:
                                A8:94:A3:2F:4C:5A:2F:AD:71:D3:1B:3C:A1:52:5F:76:E6:4
F:
                                63:2F:43:98:5E:36:CC:8C:FE:87:47:E4:9D:93:0B:71:99:A
8:
                                DD:95:C7:61:F2:CC:3E:C9:46:99:67:2C:AD:CB:30:62:0F:3
9:
                                45:93:98:10:4B:83:FB:50:F0:1E:38:82:11:6E:9E:93:AC:E
3:
```

```
                                    84:56:CA:61:79:6B:F3:60:84:53:BE:D1:27:7A:33:47:29:6
2:
                                    2D:A8:1D:3E:0F:CF:EF:69:5C:FB:13:7B:99:C2:88:E8:0D:0
7:
                                    DD:7F:DC:8F:3C:C9:DC:9F:A8:7F:B2:8B:29:50:E7:6F:17:A
0:
                                    E4:73:02:4C:6E:6F:13:A9:C9:79:3B:E8:38:3A:79:44:0D:C
8:
                                    9C:10:85:CE:52:E1:40:4B:5D:CE:5F:47:D0:D7:1B:AD:71:0
9:
                                    A8:94:37:65:D2:A0:4A:69:3D:ED:92:81:1E:10:AA:8F:DC:6
B:
                                    79:6F:1F:3A:D1:0C:EA:A1:22:00:05:A1:35:58:0E:6A:2E:C
7:
                                    FB:4E:B7:A8:23:97:9E:5A:1B:15:CA:15:3A:1D:01:58:83:8
2:
                                    6D:7D:AD:8F:94:BC:33:C0:17
X509v3 Basic Constraints:
    CA:                                     Yes
    Critical:                               Yes
X509v3 Subject key identifier:
    ID:                             1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:05:0
8:
                                            6F:FB
    Critical:                               No
X509v3 Authority key identifier:
    ID:                             1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:05:0
8:
                                            6F:FB
    Critical:                               No
X509v3 Key Usage:
    Usage:                                  Digital Signature
                                            Certificate Sign
                                            CRL Sign
    Critical:                               Yes
```

show crypto certificates crl

Данной командой выполнятся просмотр содержимого списка отозванных сертификатов.

Синтаксис

```
show crypto certificates crl [<NAME>]
```

Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа crl.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates crl ca_crl.pem
Version:                                    2
Issuer name:
    C(countryName):                         RU
    ST(stateOrProvinceName):                Siberia
    L(localityName):                        Moscow
    O(organizationName):                    Rusteletech
    OU(organizationalUnitName):             Rusteletech SC
    CN(commonName):                         ca.test.loc
    emailAddress(emailAddress):             ca@test.loc
Update info:
    Last update:                            2024-06-24 08:32:32
    Next update:                            2024-07-24 08:32:32
Signature:
    Algorithm:                              sha256WithRSAEncryption
    Value:                      9A:93:8A:CE:59:91:7B:1B:19:59:72:0F:3B:A5:45:F3:D8:17:
                                A5:8B:5C:AF:15:C9:2A:BE:25:1E:A2:67:E1:22:52:24:45:C2:
                                FD:EC:9A:ED:6E:C2:1B:00:D7:2B:25:FB:50:EE:7F:F9:9F:23:
                                FC:FA:57:3D:D3:1F:50:B7:04:40:22:9D:13:52:7B:B3:42:5D:
                                6E:13:AA:6A:55:E1:5A:43:56:CF:27:B1:BE:23:A6:F3:15:1A:
                                DA:81:19:02:56:68:F7:31:6C:22:81:83:FC:0A:A0:7A:06:01:
                                57:2B:AB:93:73:01:55:B4:7C:DB:CF:45:D1:52:0A:E1:73:14:
                                56:F6:55:DB:76:FA:66:2A:B9:9C:04:BD:4C:66:CC:21:97:14:
                                AB:FD:D7:A6:64:AB:D4:E1:4F:12:B5:32:83:C5:62:60:5B:85:
                                A8:09:5B:B8:A0:00:CA:94:04:70:13:CE:CA:BC:F2:68:01:25:
                                6A:63:5C:44:B4:33:92:B5:4E:26:CA:28:1D:BB:A8:54:34:DC:
                                23:80:A2:49:AE:2B:CF:97:D4:84:0F:0F:8D:21:04:65:91:DC:
                                83:19:F1:FB:BD:AF:01:6F:FA:09:CD:67:95:26:EE:68:69:72:
                                6D:40:BA:D7:2B:D0:D5:82:E7:D9:B4:76:30:DD:7B:F2:A3:C0:
                                D5:44:07:DD:8C:86:46:27:6A:07:CB:10:CB:31:F8:8A:99:9D:
                                82:EA:B5:A5:2E:FE:8F:0E:77:75:62:1B:FE:E8:B5:98:87:F9:
                                09:02:2B:3D:58:91:56:83:D2:E7:41:B6:1E:1C:0E:05:F0:6E:
                                DE:F3:6D:AD:1E:FB:10:81:E3:F0:28:5F:A6:42:C9:03:65:B6:
                                DC:85:DD:F7:97:9B:83:5D:C5:E1:FA:F0:90:9B:67:A3:0D:0F:
                                CE:0C:8F:40:D8:09:F7:12:F1:A3:D4:6B:AF:5E:D0:4E:56:3C:
                                41:5C:38:5E:8B:32:CA:B9:61:DD:26:87:BB:0C:D2:B2:2C:16:
                                F7:80:A3:E5:CA:92:E7:4B:30:DF:1F:21:58:80:72:23:09:DC:
                                AC:A9:16:0B:FF:51:C4:5F:FB:E6:ED:36:30:8C:59:80:83:5A:
                                3A:68:07:3D:B0:63:57:A2:EC:04:F1:B0:13:C1:E0:9C:8D:0E:
                                FC:50:D8:2E:6A:A2:BD:47:26:1D:BE:CB:34:6C:D4:85:CF:B9:
                                47:11:BB:E4:B3:0C:C8:2C:25:8E:1D:25:B6:6C:51:4B:3A:E5:
                                D2:50:5C:89:03:55:68:0F:7A:C6:F3:08:01:9C:47:9C:C2:29:
                                BF:22:20:6B:44:0C:DE:A1:26:C6:E8:FD:0F:19:E8:DC:45:9E:
                                6D:07:E1:D5:E9:FC:FC:AA
X509v3 Authority key identifier:
    ID:                         1D:60:BA:51:B1:16:35:83:53:E1:D4:33:46:40:93:BB:05:08:
                                6F:FB
    Critical:                               No
X509v3 CRL Number:
    Number:                                 4097
    Critical:                               No
Revoked certificates:
    Serial:                                 4097
        Revocation date:                    2024-06-24 08:32:21
```

show crypto certificates dh

Данной командой выполнятся просмотр имеющихся ключей Диффи-Хеллмана.

Синтаксис

```
show crypto certificates dh
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates dh
File
name

------------------------------------------------------------------
default_dh.pem

dh.pem
```

show crypto certigicates pfx

Командой выполнятся просмотр содержимого контейнера PKCS12.

Синтаксис

```
show crypto certificates pfx [<NAME>] [password asccii-text <PASSWORD>]
```

Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

<PASSWORD> – пароль, назначенный при генерации контейнера PKCS12. Максимальная длина строки до 64 символов.

При выполнении команды без параметра <NAME> происходит отображение всех сертификатов типа pfx.

В случае, если пароль не задан, то нужно определить только параметр <NAME>.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates pfx show_pkcs_no_pass.p12
MAC:
    Algorithm:
sha1
    Iteration:
2048
    Length:
20
    Salt length:
8
PKCS7 Encrypted data:
    Algorithm:
pbeWithSHA1And40BitRC2-CBC
    Iteration:
2048
    Certificate
bag:
        X509
attributes:
            localKeyID
3A:56:1C:DF:6D:67:39:1B:C6:12:33:D2:29:B3:6F:B7:6A:B
1:


62:25
        Version:
1
        Serial:
24:60:8C:4A:B7:7E:D7:2B:11:AB:34:98:48:6A:45:66:AB:0
F:

7C:80
        Subject
name:
            C(countryName):
RU
            ST(stateOrProvinceName):
Russia
            L(localityName):
Moscow
            O(organizationName):
Rusteletech
            CN(commonName):
rtt
        Issuer
name:
            C(countryName):
RU
            ST(stateOrProvinceName):
Russia
            L(localityName):
Moscow
            O(organizationName):
Rusteletech
            CN(commonName):
Rusteletech default certificate authority
        Validity
period:
            Valid after:
1970-01-01 00:03:26
            Invalid after:
2069-12-07 00:03:26
```

```
        Signature:

            Algorithm:
sha256WithRSAEncryption
            Value:
AA:B4:6C:4A:48:E6:64:62:D9:2A:05:4D:3D:7B:F7:70:0D:6
5:

BB:85:D0:BB:08:A3:75:EB:B9:B1:D1:89:B7:CD:6F:4C:55:7
A:

81:F3:7C:F2:E2:DC:68:DD:ED:4B:05:EC:9B:5A:E9:DA:DF:A
D:

AA:28:0B:BB:49:3E:E6:73:29:09:20:35:E4:A2:A8:64:10:1
A:

C8:9B:BB:3A:60:8F:10:65:E7:EE:99:E7:C1:D9:6E:4F:61:7
8:

1D:35:A4:BD:15:7C:78:4F:6C:66:B9:4E:7A:C1:08:8F:BD:8
4:

9B:11:D1:A9:EB:3A:0E:8E:FD:97:E8:2F:0E:D6:2D:DD:9F:8
2:

76:DA:FD:02:49:E9:F8:D4:D8:0A:16:D8:28:29:4F:9A:5A:3
6:

26:2A:64:DF:3C:89:6C:EE:CD:67:45:AB:F0:D9:1F:B9:FE:8
E:

0A:C3:73:32:C6:42:55:14:B9:A8:81:CE:ED:E4:9B:4B:9B:4
9:

21:1F:7E:22:FE:36:0D:1D:86:D1:18:20:CD:66:C6:18:50:7
B:

71:85:04:46:A4:CB:E5:2A:E2:10:F0:A1:12:81:9A:BD:B1:7
D:

42:84:91:E8:BD:06:77:29:30:7C:21:FF:0F:98:F1:37:3D:C
E:

5F:DC:41:3B:12:D0:CA:74:85:DB:00:18:BC:FD:F0:1B:11:9
B:

C2:21:22:2B
        Public key
info:
            Algorithm:
RSA
            Key size:
2048
            Exponent:
65537
            Modulus:
00:96:26:17:9E:E9:03:B9:03:43:16:6A:56:43:81:F2:FB:D
F:

D9:F6:A1:55:9A:3A:6D:8D:84:43:31:89:CD:22:B7:C8:4E:6
1:

49:E4:3C:A0:95:3B:D4:CE:2B:93:58:5E:EE:22:26:7A:4C:A
```

```
2:

11:EA:12:87:79:7E:E0:CF:F9:BA:B3:71:DE:C4:83:4A:84:D
0:

50:7E:B0:94:EF:D3:49:B0:9F:EA:37:F5:B3:E4:10:15:C9:7
F:

2B:B6:6C:83:A5:8E:BE:0B:E3:C3:59:1A:A3:18:B2:95:E7:3
0:

15:DC:33:C7:39:48:6D:6A:FD:97:0E:17:71:BB:88:12:FE:2
0:

B8:EC:96:06:9B:5E:BC:0A:F6:73:BF:BE:45:17:F2:30:C4:E
3:

2E:F6:87:D3:03:3E:3E:DC:3E:4C:EC:3D:9F:13:C7:57:56:A
2:

BF:06:DD:12:D3:A3:AD:FE:AC:17:87:5D:D2:0B:7C:88:3F:6
A:

39:24:E9:A9:BF:C1:07:9C:ED:8B:98:D7:43:C7:D7:9F:80:B
D:

ED:9B:24:B2:AC:0C:A7:AD:AA:B0:2D:C9:8B:38:3C:A9:70:E
D:

5F:1E:C0:14:24:B5:CB:EF:CF:3C:46:63:A2:D1:97:EB:48:0
9:

BD:3C:1E:A8:D2:A2:91:25:6A:14:6C:BF:57:A1:E7:CB:67:2
C:

B3:62:FC:B7:C1
    Certificate
bag:
        Version:
3
        Serial:
44:55:FB:9A:76:2A:2A:44:EA:90:1F:43:5F:B3:44:5C:44:9
C:

2A:02
        Subject
name:
            C(countryName):
RU
            ST(stateOrProvinceName):
Russia
            L(localityName):
Moscow
            O(organizationName):
Rusteletech
            CN(commonName):
Rusteletech default certificate authority
        Issuer
name:
            C(countryName):
RU
            ST(stateOrProvinceName):
Russia
            L(localityName):
Moscow
```

```
            O(organizationName):
Rusteletech
            CN(commonName):
Rusteletech default certificate authority
        Validity
period:
            Valid after:
1970-01-01 00:03:24
            Invalid after:
2069-12-07 00:03:24
        Signature:

            Algorithm:
sha256WithRSAEncryption
            Value:
95:42:AC:62:99:71:D8:86:53:13:E6:90:60:21:40:48:99:3
F:

8B:BC:F9:9D:A7:DC:50:1B:18:97:12:DB:75:88:5A:19:68:3
5:

8F:E7:71:16:40:04:18:D2:95:45:68:25:29:39:CD:FC:03:2
E:

B2:BE:DA:E7:A0:09:92:A2:11:E1:3E:63:E0:D5:16:8D:4C:A
7:

99:3D:D6:0E:77:11:AD:5C:14:01:A2:99:9B:24:F2:BF:34:B
0:

81:DF:85:0C:A5:D1:B8:32:77:83:0F:4B:EF:AD:E3:8F:97:5
0:

F9:B4:51:CC:EB:98:E9:CC:4A:36:AC:0A:3E:32:6B:13:4B:8
8:

31:0D:6A:E8:65:84:62:86:FD:6F:E0:71:8D:18:A3:5F:E8:C
3:

C6:C1:62:3C:2E:1D:A4:B8:5C:35:A9:BD:BA:76:A2:AA:58:8
F:

87:85:DF:41:A7:A0:CE:49:67:DD:C8:4B:0F:A0:CF:19:F5:E
7:

BB:A5:D1:8F:68:44:35:0E:49:BC:D1:DE:C6:F7:1C:8B:43:7
C:

20:12:04:14:F7:FB:23:FC:C8:4E:32:B6:76:CA:C6:85:0D:F
6:

C7:61:8F:32:AF:B0:D1:4C:2B:89:83:C4:2F:5D:85:C7:AC:B
B:

25:4F:22:6F:7B:28:25:A8:0E:2B:74:CC:E4:77:C4:F3:F9:9
D:

E7:F9:68:DB
        Public key
info:
            Algorithm:
RSA
            Key size:
2048
            Exponent:
```

```
65537
            Modulus:
00:AD:FA:04:70:35:30:4A:8A:67:32:72:09:DC:D6:6F:2B:6
3:

A3:BB:35:DC:7A:17:6D:3C:84:AA:24:A8:7C:DE:0F:21:B7:0
6:

35:AC:C7:6F:91:34:B4:04:EB:99:67:15:3B:A6:A0:E7:F6:0
3:

EF:D5:5D:38:63:3A:52:4B:23:50:68:03:B7:16:A9:99:D4:2
D:

2A:54:98:2D:98:7C:9F:D9:54:C3:44:C1:6A:09:EF:2A:21:D
5:

55:A4:0B:63:65:95:CD:4A:75:6A:AC:44:A4:C8:3F:D0:A5:B
8:

AA:F6:03:01:6C:94:B7:26:FC:AC:CE:4E:3A:B1:0B:B6:E7:F
8:

66:3B:FA:B1:DA:07:61:77:D5:0C:38:80:CF:6C:38:B3:8E:6
A:

1B:76:A7:10:AE:35:41:B7:C5:48:FA:21:27:68:8A:4D:BD:4
9:

E4:61:9B:11:D9:4F:F2:A1:0B:AF:49:53:07:C8:49:21:E1:D
3:

56:09:85:77:72:DC:A6:EB:4C:C5:C2:A6:2C:38:5B:79:34:F
5:

03:3A:16:C3:6C:16:7E:89:E5:FC:11:AB:D6:C8:03:B4:05:C
F:

BF:15:A5:C9:62:61:58:07:67:52:72:57:4D:28:C5:0D:E4:D
B:

15:95:A4:0E:D3:FF:55:E2:21:53:FF:0F:1F:AB:D0:7A:23:6
A:


21:8D:47:3A:29
        X509v3 Subject key
identifier:
            ID:
03:CC:9D:E6:A2:F0:2F:11:E6:18:48:DE:FF:C9:A4:27:54:5
B:

3F:D7
            Critical:
No
        X509v3 Authority key
identifier:
            ID:
03:CC:9D:E6:A2:F0:2F:11:E6:18:48:DE:FF:C9:A4:27:54:5
B:

3F:D7
            Critical:
No
```

```
        X509v3 Basic
Constraints:
            CA:
Yes
            Critical:
Yes
PKCS7
Data:

    Shrouded
Keybag:
        X509
attributes:
            localKeyID
3A:56:1C:DF:6D:67:39:1B:C6:12:33:D2:29:B3:6F:B7:6A:B
1:

62:25
        Algorithm:
pbeWithSHA1And3-KeyTripleDES-CBC
        Iteration:
2048
        Key algorithm:
RSA
        Key size:
2048
```

show crypto certificates private-key

Данной командой выполнятся просмотр содержимого приватного ключа.

Синтаксис

```
show crypto certificates private-key [<NAME>]
```

Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа private-key.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates private-key server.key
Key info:
    File name:                          server.key
    Issuer:                             Private key: (2048 bit)
```

show crypto certificates public-key

Командой выполнятся просмотр содержимого публичного ключа.

Синтаксис

```
show crypto certificates public-key [<NAME>]
```

Параметры

<NAME> – имя сертификата или ключа выбранного типа, задается строкой до 31 символа.

При выполнении команды без параметра происходит отображение всех сертификатов типа public-key.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates public-key pubkey.pem
File name:              pubkey.pem
Algorithm:             -
Length:                   256
Location:               -
Fingerprint:           -
```

show crypto certificates ta

Командой выполнятся просмотр имеющихся HMAC-ключей.

Синтаксис

```
show crypto certificates ta
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto certificates ta
File
name

----------------------------------------------------------------------
ta.key
```

20.2. Конфигурирование и мониторинг туннелей

access-addresses

Команда указывает список IP-адресов, которым будет разрешено находиться внутри туннеля.

Синтаксис

```
access-addresses { address-range <ADDR>[-<ADDR>] | prefix <ADDR/LEN> | object-group
<OBJ-GROUP-NETWORK-NAME> }
no access-addresses
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов, с которых разрешено принимать пакеты в этом peer. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255];

prefix <ADDR/LEN> – IP-подсеть, с которой разрешено принимать пакеты в этом peer. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32];

object-group <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-wireguard-peer)# access-addresses address-range 192.168.0.1-192.168.0.10
```

auth-nocache

Данной командой отключается кэширование пароля пользователя OpenVPN-клиента.

Использование отрицательной формы команды (no) активирует кэширование пароля пользователя OpenVPN-клиента.

Синтаксис

```
[no] auth-nocache
```

Параметры

Отсутствуют.

Значение по умолчанию

Кэширование разрешено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config)# auth-nocache
```

authentication algorithm

Данной командой устанавливается алгоритм аутентификации, который используется для аутентификации при подключении к OpenVPN-серверу.

Использование отрицательной формы команды (no) удаляет алгоритм аутентификации.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2, sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1, sha-224, rsa-sha-224, sha-256, rsa-sha-256, sha-384, rsa-sha-384, sha-512, rsa-sha-512, whirlpool.

Значение по умолчанию

Не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# authentication algorithm md5
```

authentication method

Данной командой определяется метод аутентификации, который будет использоваться при установлении удаленного подключения клиентами PPPoE, PPTP и L2TP.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

Синтаксис

```
authentication method <METHOD>
no authentication method <METHOD>
```

Параметры

<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap.

Значение по умолчанию

chap

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

Пример

```
rtt(config-pppoe)# authentication method mschap-v2
```

authentication required disable

Данной командой отключается обязательная аутентификация PPP на сервере PPTP и L2TP.

Использование отрицательной формы команды (no) включается аутентификация PPP на сервере PPTP и L2TP.

Синтаксис

```
[no] authentication required disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

Пример

```
rtt(config-pppoe)# authentication required disable
```

crypto

Данной командой указываются необходимые сертификаты для подключения к OpenVPN-серверу.

Использование отрицательной формы команды (no) удаляет название сертификата из конфигурации.

Синтаксис

```
crypto <CERTIFICATE-TYPE> <NAME>
no crypto <CERTIFICATE-TYPE>
```

Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат центра сертификации;
- cert – сертификат клиента;

- private-key – клиентский ключ;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хеллмана;
- ta – HMAC-ключ.

<NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# crypto ca KEY
```

crypto pfx

Данной командой указывается контейнер PKCS12.

Контейнер в обязательном порядке должен включать в себя сертификат удостоверяющего центра, сертификат x.509, выпущенный этим удостоверяющим центром и приватный ключ этого сертификата. Контейнер может содержать цепочку из сертификатов промежуточных удостоверяющих центров.

Использование команд **crypto ca**, **crypto cert**, **crypto private-key** и **crypto pfx** является взаимоисключающим.

Синтаксис

```
crypto pfx <NAME> [password ascii-text <PASSWORD>]
no crypto pfx <NAME>
```

Параметры

<NAME> – имя PKCS12-контейнера, задаётся строкой до 31 символа.

<PASSWORD> – пароль от PKCS12-контейнера.

Необходимый уровень привилегий

15

Командный режим

Пример

```
rtt(config-openvpn-server)# crypto pfx pkcs_openvpn_server.p12 password ascii-text
password
```

clear ip nhrp peers

Данная команда используется для очистки записей о NHRP-соседях с маршрутизатора.

Синтаксис

```
clear ip nhrp peers [ { connected | group | lower-up | nat | nhs | protected | qos
| unique | used } ] [ { nbma-address <ADDR> | tunnel gre <ID> | tunnel-address
<ADDR> } ] [ type { cached | dynamic | incomplete | negative | static } ] [ vrf
<VRF> ]
```

Параметры

connected – очистить записи, для которых разрешена отправка пользовательских пакетов через GRE-туннель;

group – очистить записи, в которых NHRP-сосед сообщил свою NHRP-группу;

lower-up – очистить записи, в которых оперативное состояние GRE-туннеля UP;

nat – очистить записи, в которых NHRP-сосед расположен за NAT;

nhs – очистить записи, в которых осуществляется регистрация на NHRP-сервере;

protected – очистить записи, в которых GRE-туннель до NHRP-соседа защищен шифрованием IPsec;

qos – очистить записи, в которых к GRE-туннелю до NHRP-соседа применена QoS-политика на основании NHRP-группы;

unique – очистить записи, для которых требуется уникальность соответствия туннельного и NBMA-адреса;

used – очистить записи, в которых через GRE-туннель проходит пользовательский трафик;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

cached – очистить записи, соответствующие GRE-туннелям между NHRP-клиентами;

dynamic – очистить записи, соответствующие GRE-туннелям от NHRP-клиента до NHRP-сервера;

incomplete – очистить незавершенные записи, в которых известен туннельный адрес, но не определен NBMA-адрес;

negative – очистить ошибочные записи, для которых ни один доступный NHRP-сервер не имеет записи о запрошенном туннельном адресе;

static – очистить записи, заданные через конфигурацию маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip nhrp peers tunnel-address 172.16.0.11
```

clear ip nhrp shortcut-routes

Данная команда служит для очистки созданных shortcut-route записей до подсетей, находящихся за NHRP-клиентами.

Синтаксис

```
clear ip nhrp shortcut-routes [ { network <ADDR/LEN> | nexthop <ADDR> | tunnel gre
<ID>  } ] [ vrf <VRF> ]
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip nhrp shortcut-routes network 1.0.0.11/32
```

clear tunnels counters

Данной командой осуществляется сброс счетчиков заданного туннеля или группы туннелей.

Синтаксис

```
clear tunnels counters [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Можно указать несколько туннелей перечислением через запятую «,» либо указать диапазон туннелей через дефис «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear tunnels counters dypseudowire 2 192.0.2.10
```

clear tunnels softgre

Данная команда разрушает softgre туннели/туннель.

Синтаксис

```
clear tunnels softgre [ remote-address <REMOTE-IP> ]
```

Параметры

<REMOTE-IP> – удаленный IP-адрес, с которого поднят softgre-туннель.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear tunnels softgre
rtt# clear tunnels softgre remote-address10.10.42.10
```

compression

Данной командой включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN.

Использование отрицательной формы команды (no) отключает механизм сжатия передаваемых данных.

Синтаксис

```
[no] compression
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# compression
```

default-profile

Данная команда позволяет использовать конфигурацию данного SoftGRE-туннеля для автоматического создания туннелей с такими же mode и local address.

Использование отрицательной формы команды (no) запрещает использование конфигурации туннеля для автоматического создания туннелей.

Синтаксис

```
[no] default-profile
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SOFTGRE

Пример

```
rtt(config-softgre)# default-profile
```

description

Данная команда используется для изменения описания конфигурируемого туннеля.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-SUBTUNNEL

CONFIG-GRE

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-WIREGUARD

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-gre)# description "tunnel to branch"
```

dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке инкапсулирующего пакета.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
dscp <DSCP>
no dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

Наследуется от инкапсулируемого пакета.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

Пример

```
rtt(config-ip4ip4)# dscp 40
```

enable

Данной командой включается туннель.

Использование отрицательной формы команды (no) отключает туннель.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Туннель выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-WIREGUARD

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-gre)# enable
```

encryption algorithm

Данной командой выбирается алгоритм шифрования, используемый при передачи данных.

Использование отрицательной формы команды (no) отключает шифрование.

Синтаксис

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Параметры

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

Значение по умолчанию

Шифрование отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# encryption algorithm aes128
```

history statistics

Данной командой включается запись статистики использования текущего туннеля.

Использование отрицательной формы команды (no) отключает запись статистики использования текущего туннеля.

Синтаксис

```
[no] history statistics
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-SUBTUNNEL

CONFIG-IP4IP4

CONFIG-L2TPV3

CONFIG-LT

CONFIG-VTI

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-L2TP

Пример

```
rtt(config-ip4ip4)# history statistics
```

ignore nameserver

Данная команда включает режим, в котором туннель игнорирует полученный от сервера DNS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ignore nameserver
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

DNS-сервер, полученный через туннель, устанавливается в таблицу dns servers.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPTP

CONFIG-PPPOE

Пример

```
rtt(config-openvpn)# ignore nameserver
```

ignore-default-route

Данная команда включает режим, в котором маршрут по умолчанию, полученный от сервера, не устанавливается в таблицу маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ignore-default-route
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Маршрут по умолчанию, полученный от сервера, устанавливается в таблицу маршрутизации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

CONFIG-PPTP

CONFIG-PPPOE

Пример

```
rtt(config-pptp)# ignore-default-route
```

interface

Данной командой определяется интерфейс, через который будет устанавливаться PPPoE-соединение.

Использование отрицательной формы команды (no) удаляет указанный интерфейс.

Синтаксис

```
interface <IF>
no interface
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPPOE

Пример

```
rtt(config-pppoe)# interface gigabitethernet 1/0/5.100
```

ip dont-fragment-bit ignore

Данной командой включается безусловная фрагментация GRE-трафика. В результате, если размер получившегося после инкапсуляции в GRE пакета больше MTU исходящего интерфейса, то GRE-пакет будет фрагментирован. В фрагментированном GRE-пакете DF-бит будет сброшен.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip dont-fragment-bit ignore
```

Параметры

Команда не содержит параметров.

Значение по умолчанию.

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip dont-fragment-bit ignore
```

ip nhrp attribute group

Данная команда задаёт имя NHRP-группы, которое будет передаваться NHRP-соседям в NHRP-сообщениях.

Использование отрицательной формы команды (no) отключает рассылку NHRP-группы.

Синтаксис

```
ip nhrp attribute group <WORD>
[no] ip nhrp attribute group
```

Параметры

<WORD> – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [^#].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp attribute group rtt-spoke-5mbps
```

ip nhrp authentication

Данная команда включает аутентификацию для протокола NHRP. У всех участников NHRP-процесса должен быть одинаковый пароль.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
ip nhrp authentication { <WORD> | encrypted <ENCRYPTED-TEXT> }
[no] ip nhrp authentication
```

Параметры

<WORD> – пароль в открытой форме, задается строкой [1..8] символов, может включать символы [0-9a-fA-F].

<ENCRYPTED-TEXT> – зашифрованный пароль, размером [1..8] байт, задаётся строкой [2..16] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp authentication password
```

ip nhrp enable

Данной командой включается работа протокола NHRP на GRE-туннеле маршрутизатора.

Использование отрицательной формы команды (no) отключает работу протокола NHRP на GRE-туннеле маршрутизатора.

Синтаксис

```
[no] ip nhrp enable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp enable
```

ip nhrp holding-time

Данная команда служит для установки времени жизни NHRP-записи, которую будет сформирована NHRP-сервере после успешной регистрации NHRP-клиента. Аналогичным образом это время будет использоваться в записи на NHRP-клиенте после успешного построения туннеля между NHRP-клиентами. Также эта команда влияет на период повторных запросов регистрации на NHRP-сервере, он рассчитывается как 1/3 от значения команды *ip nhrp holding-time*.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip nhrp holding-time <TIME>
[no] ip nhrp holding-time
```

Параметры

<TIME> – время в секундах, в течение которого у NHRP-соседа будет существовать запись о данном NHRP-клиенте, принимает значения [1..65535].

Значение по умолчанию

7200 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp holding-time 300
```

ip nhrp ipsec

Данной командой указывается имя IPsec VPN, который будет использован в качестве транспорта для GRE-туннеля с настроенным протоколом NHRP.

Использование отрицательной формы команды (no) отключает использование IPsec VPN на GRE-туннеле.

Синтаксис

```
ip nhrp ipsec <WORD> { static | dynamic }
no ip nhrp ipsec <WORD> { static | dynamic }
```

Параметры

<WORD> – имя VPN, задаётся строкой до 31 символа;

static – статическое соединение, применяется для связи с NHRP-сервером;

dynamic – динамически устанавливающееся соединение, настраивается для возможности поднятия туннелей до NHRP-клиентов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp ipsec VPN static
```

ip nhrp map

Данная команда задаёт соответствие «внутреннего» туннельного адреса с «внешним» NBMA-адресом.

Использование отрицательной формы команды (no) удаляет соответствие.

Синтаксис

```
[no] ip nhrp map <ADDR-IN> <ADDR-OUT>
```

Параметры

<ADDR-IN> – туннельный IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<ADDR-OUT> – NBMA IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp map 172.16.0.2 67.86.141.231
```

ip nhrp map group

Данная команда задаёт соответствие NHRP-группы, полученной от NHRP-соседа в сообщениях NHRP и политики QoS, которая будет применена к исходящему в сторону этого NHC-соседа трафика.

Использование отрицательной формы команды (no) удаляет соответствие.

Синтаксис

```
[no] ip nhrp map group <GROUP> service-policy output <POLICY>
```

Параметры

<GROUP> – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [^#];

<POLICY> – имя QoS-политики, задается строкой [1..31] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp map group SPOKE-5M service-policy output POLICY-5M
```

ip nhrp multicast

Данная команда определяет адресатов для отправки мультикастного трафика.

Использование отрицательной формы команды (no) удаляет адресата.

Синтаксис

```
[no] ip nhrp multicast { dynamic | nhs | <ADDR> }
```

Параметры

dynamic – отправляет трафик всем NHRP-соседям, которые установили соединение с текущим хостом через процедуру регистрации;

nhs – отправляет трафик на заданные в конфигурации NHRP-сервера, на которых успешно пройдена регистрация;

<ADDR> – отправляет трафик на заданный туннельный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp multicast nhs
```

ip nhrp nhs

Данная команда служит для указания туннельного адреса NHRP-сервера, к которому после успешной регистрации можно отправлять запросы на поиск NHRP-соседей.

Использование отрицательной формы команды (no) удаляет запись о сервере.

Синтаксис

```
ip nhrp nhs <ADDR>
no ip nhrp nhs <ADDR>
```

Параметры

<ADDR> – туннельный IP-адрес NHRP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть AAA – DDD принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp nhs 172.16.0.2
```

ip nhrp redirect

Данной командой включается механизм, который позволяет NHRP-серверу отслеживать не оптимальность прохождения трафика между NHRP-соседями. Если NHRP-сервер получает пакет от одного NHRP-соседа и пересылает его другому NHRP-соседу – то отправителю данного пакета будет отправлено NHRP-сообщение Traffic Indication.

Использование отрицательной формы команды (no) отключает данный механизм.

Синтаксис

```
[no] ip nhrp redirect
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp redirect
```

ip nhrp shortcut

Данной командой включается механизм, позволяющий обрабатывать NHRP-сообщения Traffic Indication. Если такое сообщение получено от NHRP-сервера на отправленный раннее пакет, то NHRP-клиент начет процедуру поиска NHRP-соседа, за которым доступен адрес назначения в отправленном пакете. Успешный поиск приведет к созданию туннеля между NHRP-соседями для оптимального прохождения трафика.

Использование отрицательной формы команды (no) отключает данный механизм.

Синтаксис

```
[no] ip nhrp shortcut
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip nhrp shortcut
```

ip path-mtu-discovery discovery disable

Данной командой отключается функция Path MTU Discovery. В результате, если каждое из следующих условий будет выполнено, то произойдет фрагментация GRE-пакета:

- Размер получившегося после инкапсуляции в GRE пакета больше MTU исходящего интерфейса;
- DF bit (don't fragment bit) исходного пакета не установлен.

В фрагментированном GRE-пакете DF-бит будет унаследован от оригинального пакета.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip path-mtu-discovery disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию.

Включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip path-mtu-discovery disable
```

ip tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

Синтаксис

```
ip tcp adjust-mss <MSS>
no ip tcp adjust-mss
```

Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

Значение по умолчанию

1460

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-VTI

CONFIG-LT

Пример

```
rtt(config-gre)# ip tcp adjust-mss 1400
```

ipsec authentication method

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся командой *ipsec authentication pre-shared-key* (см. раздел ipsec authentication pre-shared-key).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipsec authentication method pre-shared-key
no ipsec authentication method
```

Параметры

pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp-server)# ipsec authentication method psk
```

ipsec authentication pre-shared-key

Данной командой устанавливается общий секретный ключ для аутентификации, который должен совпадать у обоих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

Синтаксис

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-
TEXT> }| hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
no ipsec authentication pre-shared-key
```

Параметры

<TEXT> – строка [1..64] ASCII-символов.

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

ipsec ike proposal

Данной командой для L2TP-клиента назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IKE.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IKE.

Синтаксис

```
[no] ipsec ike proposal <NAME>
```

Параметры

<NAME> – имя ранее созданного профиля протокола IKE, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp)# ipsec ike proposal IKE_PROPOSAL
```

ipsec proposal

Данной командой для L2TP-клиента назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IPsec.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IPsec.

Синтаксис

```
[no] ipsec ike proposal <NAME>
```

Параметры

<NAME> – имя ранее созданного профиля IPsec, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp)# ipsec  proposal IPSEC_PROPOSAL
```

keepalive dhcp dependent-interface

Данной командой включается механизм перезапроса IP-адресов по протоколу DHCP на указанных интерфейсах при отключении GRE-туннеля по keepalive. Возможно указать до 8 интерфейсов для каждого GRE-туннеля.

Использование отрицательной формы команды (no) отключает механизм перезапроса IP-адресов по протоколу DHCP.

Синтаксис

```
keepalive dhcp dependent-interface <IF>
no keepalive dst-address
```

Параметры

<IF> – физический или агрегированный интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive dhcp dependent-interface gi 1/0/1
```

keepalive dhcp link-timeout

Данной командой указывается интервал времени между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах, указанных командой keepalive dhcp dependent-interface (см. раздел keepalive dhcp dependent-interface).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
keepalive dhcp link-timeout <SEC>
no keepalive dhcp link-timeout
```

Параметры

<SEC> – интервал в секундах между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах, указанных командой *keepalive dhcp dependent-interface*, принимает значения [1..32767] секунд.

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive dhcp link-timeout 90
```

keepalive dst-address

Данной командой настраивается IP-адрес для отправки keepalive-пакетов для проверки работоспособности туннеля. Если указанный IP-адрес недоступен, то туннель меняет оперативное состояние на DOWN. Данный параметр имеет значение только при включенном механизме keepalive (см. раздел keepalive enable).

Использование отрицательной формы команды (no) отключает данную проверку.

Синтаксис

```
keepalive dst-address <ADDR>
no keepalive dst-address
```

Параметры

<ADDR> – IP-адрес для проверки работоспособности GRE-туннеля.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive dst-address 192.168.1.57
```

keepalive enable

Данной командой включается проверка доступности удаленного шлюза туннеля. Если удаленный шлюз туннеля недоступен, то туннель меняет оперативное состояние на DOWN.

Использование отрицательной формы команды (no) отключает данную проверку.

Синтаксис

```
[no] keepalive enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию.

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive enable
```

keepalive retries

Данная команда определяет количество попыток проверки доступности удаленного шлюза туннеля. По достижению указанного количества неудачных попыток, туннель будет считаться неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
keepalive retries <VALUE>
no keepalive retries
```

Параметры

<VALUE> – количество попыток, принимает значения в диапазоне [1..255].

Значение по умолчанию

6

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive retries 8
```

keepalive timeout

Данной командой регулируется период отправки keepalive-пакетов встречной стороне.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
keepalive timeout <TIME>
no keepalive timeout
```

Параметры

<TIME> – время в секундах, принимает значения в диапазоне [1..32767].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-gre)# keepalive timeout 18
```

keepalive timeout ipsec

Данной командой задается время, которое отводится на восстановление соединения IPsec VPN, по истечении которого маршрутизатор будет перезагружен. Данный функционал работает, только когда маршрутизатор работает в режиме OTT. Режим работы OTT включается при производстве устройства.

При использовании отрицательной формы команды (no) значение приводится к дефолтному значению 180.

Синтаксис

```
keepalive timeout ipsec <TIME>
no keepalive timeout ipsec
```

Параметры

<TIME> – время в секундах, принимает значения в диапазоне [30-32767].

Значение по умолчанию

180

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# keepalive timeout ipsec 1000
```

key

Данная команда разрешает передачу ключа (key) в туннельном заголовке GRE (в соответствии с RFC 2890) и устанавливает значение ключа. Ключ может быть использован для идентификации потоков трафика в GRE-туннеле.

Использование отрицательной формы команды (no) запрещает передачу ключа.

Синтаксис

```
key <KEY>
no key
```

Параметры

<KEY> – значение KEY, принимает значения в диапазоне [1..4294967295].

Значение по умолчанию

Ключ не передаётся.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# key 40
```

load-average

Данной командой устанавливается интервал времени, за который усредняется статистика о нагрузке на туннеле.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
load-average <TIME>
no load-average
```

Параметры

<TIME> – интервал в секундах, принимает значения [5..150].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

CONFIG-SUBTUNNEL

CONFIG-L2TPv3

CONFIG-VTI

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-L2TP

Пример

```
rtt(config-gre)# load-average
```

local address

Данной командой устанавливается IP-адрес локального шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес локального шлюза.

Синтаксис

```
local address <ADDR>
no local address
```

Параметры

<ADDR> – IP-адрес локального шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

CONFIG-VTI

Пример

```
rtt(config-ip4ip4)# local address 192.168.1.1
```

local address xauth

Данной командой задается использование адреса, выдаваемого по mode config, при использовании ранее настроенного IPsec VPN в режиме XAUTH-клиента.

При использовании отрицательной формы команды (no) удаляет настройку.

Синтаксис

```
local address xauth <NAME>
no local address
```

Параметры

<NAME> – имя ранее созданного IPsec VPN, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# local address xauth IPsecVPN
```

local checksum

Данная команда включает вычисление контрольной суммы и занесение её в GRE-заголовок отправляемых пакетов.

Использование отрицательной формы команды (no) отключает процесс вычисления и отправки контрольной суммы.

Синтаксис

```
[no] local checksum
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# local checksum
```

local cookie

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет локальный cookie.

Синтаксис

```
local cookie <COOKIE>
no local cookie
```

Параметры

<COOKIE> – значение cookie, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 или 16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# local cookie 8FB51B8FB
```

local interface

Данной командой устанавливается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза GRE-туннеля.

При использовании отрицательной формы команды (no) прекращается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза.

Синтаксис

```
local interface { <IF> | <TUN> }
no local interface
```

Параметры

<IF> – тип и идентификатор интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# local interface gigabitethernet 1/0/1
```

local port

Команда определяет локальный UDP-порт, если в качестве метода инкапсуляции был выбран UDP-протокол.

Использование отрицательной формы команды (no) удаляет локальный номер UDP-порта.

Синтаксис

```
local port <UDP>
no local port
```

Параметры

<UDP> – номер UDP-порта в диапазоне [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# local port 1501
```

local session-id

Установить локальный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет локальный идентификатор сессии.

Синтаксис

```
local session-id <SESSION-ID>
no local session-id
```

Параметры

<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# local session-id 200
```

min-frag-size

Данной командой устанавливается минимальный размер пакета (в байтах). В результате пакеты, превышающие заданный размер, будут фрагментироваться при прохождении через соответствующий multilink PPP.

Использование отрицательной формы команды (no) удаляет пороговое значение.

Синтаксис

```
min-frag-size <SIZE>
```

```
no lmin-frag-size
```

Параметры

<SIZE> – размер пакета, принимает значения в диапазоне [128..1500].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-multilink)# min-frag-size 1000
```

mode

Данной командой задается режим работы SoftGRE-туннеля.

Использование отрицательной формы команды (no) снимает установленный режим.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы туннеля, возможные значения:

- data – режим данных;
- management – режим управления.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SOFTGRE

Пример

```
rtt(config-softgre)# mode data
```

mode

Данной командой указывается режим инкапсуляции для GRE-туннеля.

Использование отрицательной формы команды (no) устанавливает инкапсуляцию по умолчанию.

Синтаксис

```
mode <MODE>
```

Параметры

<MODE> – режим инкапсуляции для GRE-туннеля:

- ip – инкапсуляция IP-пакетов в GRE;
- ethernet – инкапсуляция Ethernet-фреймов в GRE.

Значение по умолчанию

ip

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# mode ethernet
```

mtu

Данной командой указывается размер MTU (Maximum Transmition Unit) для туннелей.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

```
mtu <MTU>
no mtu
```

Параметры

<MTU> – значение MTU, принимает значения в диапазоне [552..10000] (для PPPoE-туннелей принимает значения в диапазоне [552..1500]).

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-LT

CONFIG-PPTP

CONFIG-PPPOE

CONFIG-OPENVPN

CONFIG-WIREGUARD

Пример

```
rtt(config-l2tpv3)# mtu 1400
```

multipoint

Данная команда служит для перевода туннеля в режим multipoint. В этом режиме возможно установление нескольких соединений с одного туннельного интерфейса.

Использование отрицательной формы команды (no) переводит в обычный, point-to-point режим.

Синтаксис

```
[no] multipoint
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# multipoint
```

password

Команда для установки пароля пользователя OpenVPN-сервера.

Использование отрицательной формы команды (no) удаляет пароль пользователя.

Синтаксис

```
password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Параметры

<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8..32] символов, может включать символы [0-9a-fA-F];

<ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [8..32] символов.

Пароли хранятся в конфигурации в зашифрованной форме независимо от формата, использованного при вводе команды.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# password 01234567
```

peer

Данная команда используется для перехода к настройке разрешённых туннелей.

Использование отрицательной формы команды (no) удаляет разрешённый туннель.

Синтаксис

```
[no] peer <COUNT>
```

Параметры

<COUNT> – номер соответствующего пира, принимает значения [1..16].

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD

Пример

```
rtt(config-wireguard)# peer 1
```

peer lt

Данная команда используется для задания удаленной стороны (в другом VRF) логического туннеля.

Использование отрицательной формы команды (no) удаляет привязку удаленной стороны туннеля.

Синтаксис

```
[no] peer lt <ID>
```

Параметры

<ID> – идентификатор удаленной стороны логического туннеля.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LT

Пример

```
rtt(config-lt)# peer lt 2
```

port

Данной командой определяется номер UDP-порта, по которому устанавливается соединение с L2TP-сервером.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер UDP-порта, задаётся в диапазоне [1024..65535].

Значение по умолчанию

1701

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp)# port 1048
```

port

Данной командой определяется номер UDP-порта, по которому устанавливается соединение с WireGuard-сервером.

Использование отрицательной формы команды (no) удаляет указанный метод аутентификации.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер UDP-порта, задаётся в диапазоне [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIREGUARD

Пример

```
rtt(config-wireguard)# port 43020
```

ppp failure-count

Данной командой устанавливается количество неудачных data-link тестов перед разрывом сессии.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ppp failure-count <NUM>
no ppp failure-count
```

Параметры

<NUM> – количество неудачных data-link тестов, задается в диапазоне [1..100].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-l2tp)# ppp failure-count 20
```

ppp timeout keepalive

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение.

При использовании отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ppp timeout keepalive <TIME >
no ppp timeout keepalive
```

Параметры

<TIME> – время в секундах, задается в диапазоне [1..32767].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-l2tp)# ppp timeout keepalive 5000
```

pre-shared-key

Данной командой указывается необходимый симметричный ключ WireGuard.

Синтаксис

```
pre-shared-key <TYPE> <WORD>
no pre-shared-key
```

Параметры

<TYPE> – тип аргумента, устанавливаемый в качестве симметричного ключа:

- **ascii-text** – указать симметричный ключ в виде ASCII-текста, который будет сконвертирован в формат Base64;
- **base64** – указать симметричный ключ в формате Base64;
- **encrypted** – указать симметричный ключ в зашифрованном виде.

<WORD> – вводимый симметричный ключ. В зависимости от типа аргумента имеет длину 32 символа для ascii-text, 44 символа для формата base64 и 64 символа для зашифрованного вида.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-wireguard-peer)# pre-shared-key base64
hHj3XWiLpb49G5yhKYF0bYKIMhExEtRnN+1sLbWwl8s=
```

protocol

Выбор метода инкапсуляции для туннеля L2TPv3.

Синтаксис

```
protocol <TYPE>
no protocol
```

Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- IP-инкапсуляция в IP-пакет;
- UDP-инкапсуляция в UDP-дейтаграммы.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# protocol ip
```

private-key

Данной командой указывается необходимый приватный ключ клиента WireGuard.

Синтаксис

private-key <NAME>

no private-key

Параметры

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD

Пример

```
rtt(config-wireguard)# private-key client.priv
```

protocol

Выбор метода инкапсуляции для туннеля OPENVPN.

Синтаксис

```
protocol <TYPE>
no protocol
```

Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- TCP – инкапсуляция в TCP-сегмент;
- UDP – инкапсуляция в UDP-дейтаграммы.

Значение по умолчанию

TCP

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# protocol tcp
```

public-key

Данной командой указывается необходимый публичный ключ сервера WireGuard.

Синтаксис

public-key <NAME>

no public-key

Параметры

<NAME> – имя публичного ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-wireguard-peer)# public-key server.pub
```

remote address

Данной командой устанавливается IP-адрес удаленного шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

Синтаксис

```
remote address <ADDR>
no remote address
```

Параметры

<ADDR> – IP-адрес удаленного шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SOFTGRE

CONFIG-L2TP

CONFIG-L2TPV3

CONFIG-VTI

CONFIG-PPTP

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-ip4ip4)# remote address 192.168.1.2
```

remote address

Данной командой устанавливается IP-адрес и TCP/UDP-порт удаленного шлюза туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

Синтаксис

```
[no] remote address <ADDR> [ port <PORT>]
```

Параметры

<ADDR> – IP-адрес удаленного шлюза;

<PORT> – номер TCP/UDP-порта удаленного шлюза в диапазоне [1..65535].

Значение по умолчанию

<PORT> – 1194.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# remote address 192.168.1.2 port 1233
```

remote address xauth

Данной командой задается использование адреса management-ip или data-ip, выдаваемого по mode config, при использовании ранее настроенного IPsec VPN в режиме XAUTH-клиента. Требует наличия соответствующих настроек на IPsec-VPN-сервере.

При использовании отрицательной формы команды (no) удаляет настройку.

Синтаксис

```
remote address xauth <NAME> {management-ip|data-ip}
no remote address
```

Параметры

<NAME> – имя ранее созданного IPsec VPN, задаётся строкой до 31 символа;

management-ip – адрес, получаемый по mode config при установке IPsec VPN в режиме XAUTH-клиента. Требует наличия в настройках IPsec-VPN-сервера роутера ELTEX_MANAGEMENT_IP (28683).

data-ip – адрес, получаемый по mode config при установке IPsec VPN в режиме XAUTH-клиента. Требует наличия в настройках IPsec-VPN-сервера ELTEX_DATA_IP (28684).

Значение по умолчанию

Не задано.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# remote address xauth IPsecVPN
```

remote checksum

Команда включает проверку наличия и соответствия значений контрольной суммы в заголовках принимаемых GRE-пакетов.

Использование отрицательной формы команды (no) отключает проверку контрольной суммы.

Синтаксис

```
[no] remote checksum
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

По умолчанию проверка контрольной суммы выключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# remote checksum
```

remote cookie

Данная команда определяет значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией.

Использование отрицательной формы команды (no) удаляет удаленный cookie.

Синтаксис

```
remote cookie <COOKIE>
no remote cookie
```

Параметры

<COOKIE> – значение cookie, принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде [8 или 16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# remote cookie 8FB51B8FB
```

remote port

Данная команда определяет удаленный UDP-порт, если в качестве метода инкапсуляции был выбран UDP.

Использование отрицательной формы команды (no) удаляет удаленный номер UDP-порта.

Синтаксис

```
remote port <UDP>
no remote port
```

Параметры

<UDP> – номер UDP-порта в диапазоне [1..65535].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-l2tpv3)# remote port 65000
```

remote session-id

Данной командой устанавливается удаленный идентификатор сессии.

Использование отрицательной формы команды (no) удаляет удаленный идентификатор сессии.

Синтаксис

```
remote session-id <SESSION-ID>
no remote session-id
```

Параметры

<SESSION-ID> – идентификатор сессии, принимает значение в диапазоне [1..200000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2TPV3

Пример

```
rtt(config-l2tpv3)# remote session-id 2
```

route-metric

Данной командой назначается метрика маршрутов получаемых клиентом от OpenVPN-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
route-metric <METRIC>
no route-metric
```

Параметры

<METRIC> – метрика маршрута, принимает значение в диапазоне [0..255].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config)# route-metric 100
```

route-nopull

Данной командой отключается применение маршрутов, передаваемых OpenVPN-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] route-nopull
```

Параметры

Отсутствуют.

Значение по умолчанию

Игнорирование маршрутов отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config)# route-nopull
```

snmp init-trap

Данная команда используется для включения отправки snmp-trap о включении/отключении туннеля.

Использование отрицательной формы команды (no) отключает отправку snmp-trap о включении/отключении туннеля.

Синтаксис

```
[no] snmp init-trap
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-GRE

CONFIG-SUBTUNNEL

Пример

```
rtt(config-gre)# snmp init-trap
```

show ip nhrp peers

Данная команда служит для просмотра записей об NHRP-соседях.

Синтаксис

```
show ip nhrp peers [ { connected | group [ name <GROUP> ] | lower-up | nat | nhs |
protected | qos | unique | used } ] [ { nbma-address <ADDR> | tunnel gre <ID> |
tunnel-address <ADDR> } ] [ type { cached | dynamic | incomplete | negative |
static } ] [ vrf <VRF> ] [ detailed ]
```

Параметры

connected – показать записи, для которых разрешена отправка пользовательских пакетов через GRE-туннель;

group – показать записи, в которых NHRP-сосед сообщил свою NHRP-группу, при опциональном указании имени NHRP-группы будут выведены NHRP-записи с принадлежностью к указанной группе;

<GROUP> – имя NHRP-группы, задаётся строкой [1..40] символов, не принимает символы [^#];

lower-up – показать записи, в которых оперативное состояние GRE-туннеля – UP;

nat – показать записи, в которых NHRP-сосед расположен за NAT;

nhs – показать записи, в которых осуществляется регистрация на NHRP-сервере;

protected – показать записи, в которых GRE-туннель до NHRP-соседа защищен шифрованием IPsec;

qos – показать записи, в которых к GRE-туннелю до NHRP-соседа применена QoS политика на основании NHRP-группы;

unique – показать записи, для которых требуется уникальность соответствия туннельного и NBMA-адреса;

used – показать записи, в которых через GRE-туннель проходит пользовательский трафик;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

cached – показать записи, соответствующие GRE-туннелям между NHRP-клиентами;

dynamic – показать записи, соответствующие GRE-туннелям от NHRP-клиента до NHRP-сервера;

incomplete – показать незавершенные записи, в которых известен туннельный адрес, но не определен NBMA-адрес;

negative – показать ошибочные записи, для которых ни один доступный NHRP-сервер не имеет записи о запрошенном туннельном адресе;

static – показать записи, заданные через конфигурацию маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа;

detailed – показать информацию об отфильтрованных NHRP-записях в подробном виде.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show ip nhrp peers
  Flags: E - unique, R - nhs, U - used, L - lower-up
         C - connected, G - group, Q - qos, N - nat
         P - protected, X - undefined

 Tunnel address         NBMA address       Tunnel      Expire      Created          Type              Flags
                                                       (h:m:s)     (d,h:m:s)
 --------------------   ----------------   ---------   ---------   --------------   ---------------   ----------
 172.16.0.1             10.0.104.2         gre 1       --          --               static            RLC
 172.16.0.11            98.0.101.2         gre 1       01:31:56    00,00:28:03       cached            LCN
rtt# show ip nhrp peers detailed
Tunnel:                       gre 1
Type:                         static
Tunnel address:               172.16.0.1
NBMA address:                 10.0.104.2
NAT-OA address:               --
Flags:                        nhs, lower-up, connected
Created (d,h:m:s):            --
Expire (h:m:s):               --
Re-registration in (h:m:s): 00:10:56
IPSec protection:             Disabled
Group:                        --
QoS policy output:            --

Tunnel:                       gre 1
Type:                         cached
Tunnel address:               172.16.0.11
NBMA address:                 98.0.101.2
NAT-OA address:               10.0.101.2
Flags:                        lower-up, connected, nat
Created (d,h:m:s):            00,00:28:05
Expire (h:m:s):               01:31:54
Re-registration in (h:m:s): --
IPSec protection:             Disabled
Group:                        --
QoS policy output:            --
rtt#
```

show ip nhrp shortcut-routes

Данная команда служит для просмотра созданных shortcut-route записей до подсетей, находящихся за NHRP NHC.

Синтаксис

```
show ip nhrp shortcut-routes [ { network <ADDR/LEN> | nexthop <ADDR> | tunnel gre
<ID>  } ] [ vrf <VRF> ]
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор GRE-туннеля;

<VRF> – имя экземпляра VRF, задается строкой от 1 до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show ip nhrp shortcut-routes
Network                 Nexthop             Tunnel      Expire      Created
                                                        (h:m:s)     (d,h:m:s)
--------------------    ----------------    ---------   ---------   --------------
1.0.0.11/32             172.16.0.11         gre 1       01:59:35    0,00:00:24
1.0.0.12/32             172.16.0.12         gre 1       01:59:49    0,00:00:10
rtt#
```

show tunnels configuration

Командой выполняется просмотр конфигурации туннеля.

Синтаксис

```
show tunnels configuration [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] |
dypseudowire pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show tunnels configuration gre 25
State:                                              enabled
Description:
Local address:                                      14.0.0.2
Remote address:                                     14.0.0.1
Calculates checksums for outgoing GRE packets:      no
Requires that all input GRE packets were checksum: no
key:                                                -
TTL:                                                Inherit
DSCP:                                               0
MTU:                                                1500
Security zone:                                      remote
```

show tunnels counters

Командой выполняется просмотр счетчиков на туннелях.

Синтаксис

```
show tunnels counters [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены счетчики всех туннелей заданной группы. Если задан определённый туннель, то будет отображена детальная информация по данному туннелю.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show tunnels counters  dypseudowire 2 192.0.2.10
Tunnel 'dypseudowire 2_192.0.2.10' counters:
 Packets received:               1006
 Bytes received:                 84200
 Dropped on receive:             0
 Receive errors:                 0
 Multicasts received:            0
 Receive length errors:          0
 Receive buffer overflow errors: 0
```

```
Receive CRC errors:             0
Receive frame errors:           0
Receive FIFO errors:            0
Receive missed errors:          0
Receive compressed:             0
Packets transmitted:            1006
Bytes transmitted:              102308
Dropped on transmit:            0
Transmit errors:                0
Transmit aborted errors:        0
Transmit carrier errors:        0
Transmit FIFO errors:           0
Transmit heartbeat errors:      0
Transmit window errors:         0
Transmit comressed:             0
Collisions:                     0
```

show tunnels history

Команда используется для просмотра статистики использования туннеля.

Синтаксис

```
show tunnels history [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire
pseudowire [ <ID> <NEIGBOR-ADDR> ] } ] [ timer <TIMER}> ]
```

Параметры

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

<ID> – идентификатор псевдопровода/динамического псевдопровода.

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TIMER> – необязательный ключ timer. В качестве параметров для данного ключа могут выступать:

- hours – отображает историю за последние 72 часа;
- minutes – отображает историю за последние 60 минут;
- seconds – отображает историю за последние 60 секунд;
- При отсутствии ключа timer выводятся 3 таблицы истории использования туннеля/туннелей.

Необходимый уровень привилегий

5

Командный режим

ROOT

Пример

```
rtt# show tunnel history gre 1 timer minutes
```

```
gre 1
Last 60 minutes:
Timer  Recv utilization, Kbit/s  Sent utilization, Kbit/s  Recv errors  Sent errors  Output drops
------ ------------------------ ------------------------ ------------ ------------ --------------
0-1   240            16             0       0       0
1-2   961            64             0       0       0
2-3   962            64             0       0       0
3-4   962            64             0       0       0
4-5   960            64             0       0       0
5-6   961            64             0       0       0
6-7   719            64             0       0       0
7-8   960            64             0       0       0
8-9   800            65             0       0       0
9-10  962            64             0       0       0
10-11 865            64             0       0       0
11-12 962            64             0       0       0
12-13 817            65             0       0       0
13-14 962            65             0       0       0
14-15 961            65             0       0       0
15-16 880            60             0       0       0
16-17 960            63             0       0       0
17-18  0             0              0       0       0
18-19  0             0              0       0       0
19-20  0             0              0       0       0
20-21  0             0              0       0       0
21-22  0             0              0       0       0
```

show tunnels status

Команда используется для просмотра состояния системных интерфейсов.

Синтаксис

```
show tunnels status [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] | dypseudowire pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

В команде можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены статусы всех туннелей заданной группы. Если задан конкретный туннель, то будет отображена детальная информация по данному туннелю;

<ID> – идентификатор псевдопровода/динамического псевдопровода;

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show tunnels status
 Tunnel            Admin   Link    MTU      Local IP          Remote IP         Last change
                   state   state                                                (d,h:m:s)
 ---------------   -----   -----   ------   ---------------   ---------------   -------------
 vti 1             Up      Up      1500     3.3.3.2           3.3.3.1           00,00:10:01
 gre 1             Up      Up      1500     3.3.3.2           3.3.3.1           00,00:10:01
```

show tunnels utilization

Команда используется для просмотра средней нагрузки в туннелях за указанный период.

Синтаксис

```
show tunnels utilization [{ <TUN> | pseudowire [ <ID> <NEIGBOR-ADDR> ] |
dypseudowire pseudowire [ <ID> <NEIGBOR-ADDR> ] } ]
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

Можно указать несколько туннелей перечислением через «,» либо указать диапазон интерфейсов через «-». Если не указывать индексы туннелей, то будут очищены счетчики всех туннелей заданной группы;

<ID> – идентификатор псевдопровода/динамического псевдопровода;

<NEIGBOR-ADDR> – IP-адрес задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show tunnels utilization gre 2
Tunnel      Period, s     Sent,          Recv,         Frames Sent    Frames Recv
                          Kbit/s         Kbit/s
---------   -----------   ------------   ------------   ------------   -----------
gre 2       15            0              0              0              0
```

ttl

Команда задаёт значение времени жизни TTL для туннельных пакетов.

Использование отрицательной формы команды (no) устанавливает значение TTL по умолчанию.

Синтаксис

```
ttl <TTL>
no ttl
```

Параметры

<TTL> – значение TTL, принимает значения в диапазоне [1..255].

Значение по умолчанию

Наследуется от инкапсулируемого пакета.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IP4IP4

CONFIG-GRE

Пример

```
rtt(config-ip4ip4)# ttl 10
```

tunnel

Данная команда позволяет перейти в режим конфигурирования туннеля.

Использование отрицательной формы команды (no) удаляет туннель.

Синтаксис

```
[no] tunnel <TUN>
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Переход в режим конфигурирования туннеля l2tp 10:

```
rtt(config)# tunnel l2tp 10
rtt(config-l2tp)#
```

Пример 2

Переход в режим конфигурирования туннеля l2tpv3 10:

```
rtt(config)# tunnel l2tpv3 10
rtt(config-l2tpv3)#
```

Пример 3

Переход в режим конфигурирования туннеля ip4ip4 200:

```
rtt(config)# tunnel ip4ip4 200
rtt(config-ip4ip4)#
```

Пример 4

Переход в режим конфигурирования туннеля gre 25:

```
rtt(config)# tunnel gre 25
rtt(config-gre)#
```

Пример 5

Переход в режим конфигурирования туннеля vti 125:

```
rtt(config)# tunnel vti 125
rtt(config-vti)#
```

Пример 6

Переход в режим конфигурирования туннеля pptp 10:

```
rtt(config)# tunnel pptp 10
rtt(config-pptp)#
```

Пример 7

Переход в режим конфигурирования туннеля pppoe 8:

```
rtt(config)# tunnel pppoe 8
rtt(config-pppoe)#
```

tunnel

Данной командой указывается режим инкапсуляции для OpenVPN-клиента.

Использование отрицательной формы команды (no) устанавливает инкапсуляцию по умолчанию.

Синтаксис

```
tunnel <MODE>
```

Параметры

<MODE> – режим инкапсуляции для OpenVPN-клиента:

- ip – инкапсуляция IP-пакетов в OpenVPN;
- ethernet – инкапсуляция Ethernet-фреймов в OpenVPN.

Значение по умолчанию

ip

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# mode ethernet
```

tunnel-source

Данной командой указывается имя VRF, от IP-интерфейса которого будет строиться данный GRE-туннель. Данная команда актуальна в случае, если GRE-туннель строится через VRF, отличный от VRF самого туннеля.

Использование отрицательной формы команды (no) устанавливает режим, когда GRE-туннель и IP-интерфейс, от которого строится GRE-туннель, находятся в одном VRF.

Синтаксис

```
tunnel-source [ vrf <VRF> ]
[no] tunnel-source
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Без указания ключа "vrf" и имени экземпляра VRF, будет использоваться IP-интерфейс глобальной конфигурации.

Значение по умолчанию

Отключено (GRE-туннель и IP-интерфейс относятся к одному VRF).

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# tunnel-source vrf magistral
```

username

Данной командой задается пользователь и пароль для подключения к L2TP-, PPPoE- или PPTP-серверу.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Синтаксис

```
username <NAME> password ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа;

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 64] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [2..128] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-pptp)# username fedor password ascii-text password
```

username

Данной командой создается пользователь для подключения к OpenVPN-серверу.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Синтаксис

```
username <NAME>
no username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN

Пример

```
rtt(config-openvpn)# username fedor
```

20.3. Настройки IPsec VPN

20.3.1. Управление VPN. Настройки IKE

access profile

Данной командой создается профиль настроек пользователя для IKE-GATEWAY с определенным именем и осуществляется переход в режим конфигурирования профиля.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль настроек пользователя для IKE-GATEWAY.

Синтаксис

```
[no] access profile <NAME>
```

Параметры

<NAME> – имя профиля пользователя для IKE-GATEWAY, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# access profile OFFICE
```

access-profile

Данной командой указывается локальный список пользователей для авторизации. В случае настройки на IPsec VPN Remote Access сервере указывается только название локального списка пользователей, а в случае настройки IPsec VPN Remote Access клиента дополнительно указывается имя пользователя, который будет использован для авторизации на сервере.

Использование отрицательной формы команды (no) удаляет заданный профиль.

Синтаксис

```
access-profile <NAME> [ client <USER-NAME> ]
[no] access-profile <NAME>
```

Параметры

<NAME> – название локального списка пользователей, задаётся строкой до 31 символа;

<USER-NAME> – имя пользователя из прикрепленного, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gateway)# access-profile OFFICE client admin
```

address-assignment pool

Команда используется для создания пула адресов и настройки передаваемых параметров для динамической конфигурации IPsec-клиентов.

Использование отрицательной формы команды (no) удаляет пула адресов.

Синтаксис

```
[no] address-assignment pool <NAME>
```

Параметры

<NAME> – имя пула адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# address-assignment pool CENTER
rtt(config-pool)#
```

assign-interface

Данной командой указывается loopback-интерфейс для назначения динамического адреса, получаемого от IPsec-VPN-сервера.

При использовании отрицательной формы команды (no) удаляется loopback-интерфейс для назначения динамического адреса, получаемого от IPsec-VPN-сервера.

Синтаксис

```
assign-interface loopback <LOOPBACK>|<LOOPBACK>-<LOOPBACK>
no assign-interface
```

Параметры

<LOOPBACK> – номер созданного ранее loopback-интерфейса, принимает значение в диапазоне [1..8].

Значение по умолчанию

Отсутствует

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GW

Пример

```
rtt(config-ike-gw)# assign-interface loopback 3
```

authentication algorithm

Данной командой устанавливается алгоритм аутентификации, который используется для аутентификации сообщений установленного IKE-соединения. При установлении IKE-соединения используется аутентификация сообщений по ключу (authentication, см. password).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.

Значение по умолчанию

sha1

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
rtt(config-ike-proposal)# authentication algorithm md5
```

authentication mode

Данной командой устанавливается режим XAUTH аутентификации удаленных пользователей, подключающихся по протоколу IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

Синтаксис

```
authentication mode { local | radius | client }
no authentication mode
```

Параметры

local – режим аутентификации, использующий локальную базу пользователей конфигурируемого профиля;

radius – режим, при котором аутентификация пользователей проходит через RADIUS-сервер;

client – режим, используемый XAUTH-клиентом.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# authentication mode local
```

authentication method

Данной командой выбирается метод аутентификации для IKE-соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
authentication method <METHOD>
no authentication method
```

Параметры

<METHOD> – метод аутентификации IKE-сессии. Может принимать значения:

- pre-shared-key – метод аутентификации, использующий предварительно согласованные ключи, которые должны совпадать у обоих участников IKE-сессии. Ключ задается в команде pre-shared-key;
- keyring – метод аутентификации, использующий набор предварительно согласованных ключей. Набор ключей задается командой keyring;
- public-key – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей должны быть загружены в локальное хранилище маршрутизатора и указаны в конфигурации через команды crypto ca, crypto-local-crt и crypto local-crt-key;
- trustpoint – метод аутентификации, использующий приватные ключи и сертификаты X.509. Файлы сертификатов и ключей предоставляются PKI-клиентом, который автоматически выписывает актуальные сертификаты у удостоверяющего центра. Используемый PKI-клиент задается в команде crypto trustpoint;

- xauth-psk-key – метод расширенной аутентификации, использующий в качестве первого фактора аутентификации предварительно согласованные ключи и пару логин-пароль пользователя в качестве второго фактора аутентификации;
- eap – метод расширенной аутентификации, использующий приватные ключи и сертификаты X.509 для аутентификации ответчика в IKE-сессии и пару логин-пароль пользователя для аутентификации инициатора IKE-сессии.

Значение по умолчанию

pre-shared-key

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-proposal)# authentication method pre-shared-key
```

bind-interface vti

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля «route-based».

Использование отрицательной формы команды (no) удаляет привязку к туннельному интерфейсу.

Синтаксис

```
bind-interface vti <VTI>
no bind-interface vti
```

Параметры

<VTI> – идентификационный номер интерфейса VTI.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# bind-interface vti 1
```

crypto

Данной командой указываются источники приватных ключей и сертификатов для использования в установлении IKE-сессии.

Использование отрицательной формы команды (no) удаляет источник из конфигурации.

Синтаксис

```
crypto <TYPE> <NAME>
no crypto <TYPE>
```

Параметры

<TYPE> – тип источника ключевой информации, может принимать следующие значения:

- ca – имя файла сертификата удостоверяющего центра;
- crl – имя файла списка отозванных сертификатов удостоверяющего центра;
- local-crt – имя файла сертификата, используемого маршрутизатором при установлении IKE-сессии;
- local-crt-key – имя файла приватного ключа, используемого маршрутизатором при установлении IKE-сессии;
- remote-crt – имя файла сертификата, используемого маршрутизатором для аутентификации удаленного IKE-соседа;
- trustpoint – имя PKI-клиента, который отвечает за автоматическую генерацию приватного ключа и выпуск сертификата у вышестоящего удостоверяющего центра для дальнейшего использования в IKE-сессии.

<NAME> – имя файла или PKI-клиента, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# crypto ca KEY
```

data-tunnel address

Данной командой указывается IP-адрес для постройки GRE data туннеля, передаваемого клиенту, подключаемому через IPsec с использованием динамического конфигурирования параметров. GRE data туннель должен быть поддержан на стороне клиента (требует Rusteletech_DATA_IP(28684).

Использование отрицательной формы команды (no) удаляет IP-адрес для постройки GRE data туннеля.

Синтаксис

```
data-tunnel address <ADDR>
no data-tunnel address
```

Параметры

<ADDR> – IP-адрес для построения GRE data туннеля, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отсутствует

Необходимый уровень привилегий

10

Командный режим

CONFIG-POOL

Пример

```
rtt(config-pool)# data-tunnel address 192.168.2.66
```

dead-peer-detection action

Данной командой устанавливается действие, которое должно предпринять устройство в случае обнаружения недоступности IPsec-соседа механизмом Dead Peer Detection.

Dead Peer Detection (DPD) – это механизм проверки состояния и доступности соседних устройств. Механизм периодически отправляет R-U-THERE сообщения (для IKE версии 1) или пустые INFORMATIONAL сообщения (для IKE версии 2) для проверки доступности IPsec-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
dead-peer-detection action <MODE>
no dead-peer-detection action
```

Параметры

<MODE> – режим работы DPD:

- restart – соединение переустанавливается;
- clear – соединение останавливается;
- hold – соединение поддерживается;
- none – механизм выключен, никаких действий не предпринимается.

Значение по умолчанию

none

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# dead-peer-detection action clear
```

dead-peer-detection interval

Данной командой устанавливается интервал между отправкой сообщений механизмом DPD.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
dead-peer-detection interval <SEC>
no dead-peer-detection interval
```

Параметры

<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.

Значение по умолчанию

2 секунды

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# dead-peer-detection interval 15
```

dead-peer-detection timeout

Данной командой задаётся период времени ожидания ответа на сообщения механизма DPD.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
dead-peer-detection timeout <SEC>
no dead-peer-detection timeout
```

Параметры

<SEC> – период времени для ответа на сообщения механизма DPD, принимает значения [1..180] секунд.

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# dead-peer-detection timeout 60
```

description

Команда используется для изменения описания профиля, набора ключей, политики или шлюза протокола IKE.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-PROPOSAL

CONFIG-IKE-POLICY

CONFIG-IKE-KEYRING

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-proposal)# description "my proposal"
```

dh-group

Данной командой устанавливается номер группы метода Диффи-Хеллмана. Номер группы определяет уровень защищенности IKE-соединения при обмене ключами – защищенность возрастает с ростом номера группы, но увеличивается и время установления соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
dh-group <DH-GROUP>
no dh-group
```

Параметры

<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14-31].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
rtt(config-ike-proposal)# dh-group 5
```

encryption algorithm

Данной командой выбирается алгоритм шифрования, используемый при установлении IKE-соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Параметры

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-PROPOSAL

Пример

```
rtt(config-ike-proposal)# encryption algorithm aes128
```

identity

Данной командой устанавливается соответствие идентификатора IPsec-клиента и ключа PSK, который будет использован при аутентификации.

Использование отрицательной формы команды (no) удаляет соответствие для указанного идентификатора.

Синтаксис

```
identity { dns <NAME> | ipv4 <ADDR/LEN> } pre-shared-key { ascii-text { <TEXT> |
encrypted <ENCRYPTED-TEXT>} | hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
no identity { dns <NAME> | ipv4 <ADDR/LEN> }
```

Параметры

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – "router.example.loc". В команде также можно указывать wild-card домены, используя символ "*", например "*.example.loc";

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED_TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED_HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-KEYRING

Пример

```
rtt(config-ike-keyring)# identity ipv4 98.0.127.37/32 pre-shared-key ascii-text
password
rtt(config-ike-keyring)# identity dns router.example.loc pre-shared-key ascii-text
password
```

ike-policy

Данной командой устанавливается привязка политики протокола IKE к шлюзу.

Использование отрицательной формы команды (no) удаляет привязку политики.

Синтаксис

```
[no] ike-policy <NAME>
```

Параметры

<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# ike-policy ike_pol1
```

ip prefix

Данной командой указывается пул адресов, из которого адреса будут выдаваться IPsec-клиентам.

Использование отрицательной формы команды (no) удаляет пул адресов, из которого адреса будут выдаваться IPsec-клиентам.

Синтаксис

```
ip prefix <ADDR/LEN>
no ip prefix
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Значение по умолчанию

Не задан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POOL

Пример

```
rtt(config-pool)# ip prefix 192.168.0.0/16
```

keyring

Данной командой указывается набор ключей аутентификации IKE.

Использование отрицательной формы команды (no) удаляет набор ключей.

Синтаксис

```
keyring <NAME>
no keyring
```

Параметры

<NAME> – название набор ключей аутентификации IKE, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# keyring ike_keyring
```

lifetime seconds

Данной командой задаётся время жизни соединения протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
lifetime seconds <SEC>
no lifetime seconds
```

Параметры

<SEC> – период времени, принимает значения [4 ..86400] секунд.

Значение по умолчанию

10800 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# lifetime 21600
```

local address

Данной командой устанавливается IP-адрес локального шлюза IPsec-туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес локального шлюза.

Синтаксис

```
local address <ADDR>
no local address
```

Параметры

<ADDR> – IP-адрес локального шлюза.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# local address 192.168.1.1
```

local id

Данной командой устанавливается идентификатор локального шлюза IPsec-туннеля, использующийся для идентификации в процессе согласования IKE-ключей. Если команда не указана, то при использовании аутентификации по ключам (PSK) в качестве идентификатора локального шлюза используется адрес локального шлюза. При использовании аутентификации по сертификатам X.509 в качестве идентификатора локального шлюза используется значение "Distinguished name" владельца сертификата X.509.

Использование отрицательной формы команды (no) удаляет настройку идентификатора локального шлюза.

Синтаксис

```
local id { any | dns <NAME> | ipv4 <ADDR> | keyid <KEY> }
no local id
```

Параметры

any – ключ, обозначающий "любой" идентификатор;

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – router.example.loc;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<KEY> – текстовая строка длиной до 255 символов.

Необходимый уровень привилегий

10

Значение по умолчанию

Отсутствует

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# local id ipv4 192.168.18.1
```

local interface

Данной командой устанавливается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза IPsec-туннеля.

При использовании отрицательной формы команды (no) прекращается использование IP-адреса, назначенного на интерфейс в качестве локального шлюза.

Синтаксис

```
local interface <IF>
no local interface
```

Параметры

<IF> – тип и идентификатор интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GW

Пример

```
rtt(config-ike-gw)# local interface gigabitethernet 1/0/1
```

local network

Данной командой устанавливается IP-адрес подсети отправителя, а также IP-протокол и порт. Трафик, удовлетворяющий заданным критериям, будет направлен в IPsec-туннель.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети отправителя.

Синтаксис

```
[no] local network { <ADDR/LEN> | dynamic } [ protocol { <TYPE> | <ID> } [ port
<PORT> ] ]
```

Параметры

<ADDR/LEN> – IP-подсеть отправителя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

dynamic – в качестве IP-адрес подсети отправителя будет использован IP-адрес, с которого устанавливается IPsec-соединение;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# local network 192.168.1.0/24 protocol tcp port 22
```

management-tunnel address

Данной командой устанавливается IP-адрес туннеля для постройки GRE management туннеля, передаваемого клиенту, подключаемому через IPsec с использованием динамического конфигурирования параметров. GRE management туннель должен быть поддержан на стороне клиента (требует ELTEX_MANAGEMENT_IP(28683)).

Использование отрицательной формы команды (no) удаляет IP-адрес туннеля для постройки GRE management туннеля.

Синтаксис

```
management-tunnel address <ADDR>
no management-tunnel address
```

Параметры

<ADDR> –IP-адрес для постройки GRE management туннеля, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POOL

Пример

```
rtt(config-pool)# management-tunnel address 192.168.2.87
```

mobike disable

Данная команда отключает расширение MOBIKE IKEv2, которое позволяет инициатору ike-сессии изменять local address в соответствии с RFC 4555.

Использование отрицательной формы команды (no) активирует функцию автоматического выбора local address при недоступности описанного в конфигурации.

Синтаксис

```
[ no ] mobike disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gateway)# mobike disable
```

mode

Данной командой устанавливается режим согласования первой фазы протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим первой фазы IKE, принимает значения:

- main – состоит из трех двусторонних обменов между отправителем и получателем:
  - Согласуются алгоритмы аутентификации и шифрования, которые будут использоваться для защиты IKE-соединения посредством сопоставления профилей протокола IKE каждого узла;
  - Используя алгоритм Диффи-Хеллмана, стороны обмениваются общим секретным ключом. Также узлы проверяют идентификацию друг друга путем передачи и подтверждения последовательности псевдослучайных чисел;
  - Проверяется идентичность противоположной стороны. В результате выполнения основного режима создается безопасный канал для второй фазы протокола IKE.
- aggressive – этот режим обходится меньшим числом обменов и, соответственно, числом пакетов:
  - В первом сообщении (от инициатора) отправляется информация, которая используется для установления IKE-соединения: предложение параметров SA, инициирование обмена Диффи-Хеллмана, отправление псевдослучайного числа и идентификатора пакета;
  - Во втором сообщении ответчик принимает SA, аутентифицирует инициатора, отправляет псевдослучайное число и свой IKE-идентификатор;
  - В третьем сообщении инициатор аутентифицирует ответчика и подтверждает обмен.

Значение по умолчанию

main

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# mode aggressive
```

mode

Данной командой устанавливается режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим перенаправления трафика в туннель, принимает значения:

- policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям;
- route-based – трафик перенаправляется на основе маршрутов, шлюзом у которых является туннельный интерфейс.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# mode route-based
```

password

Данная команда используется для установки пароля пользователя для IKE-GETWAY. Пароль может быть задан как в открытом виде, так и в виде хеш sha512.

Использование отрицательной формы команды (no) удаляет пароль пользователя для IKE-GETWAY из системы.

Синтаксис

```
password ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [8..32] символов, принимает значения [0-9a-fA-F].

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой до 110 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROFILE

Пример

```
rtt(config-profile) password tteesstt
```

password local-crt-key

Данная команда используется для установки пароля от зашифрованной цепочки сертификатов (сертификаты назначаются при помощи команды crypto).

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
password local-crt-key ascii-text { <CLEAR-TEXT> | encrypted <HASH_SHA512> }
no password local-crt-key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [8..32] символов, принимает значения [0-9a-fA-F].

<HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой до 110 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy) password tteesstt
```

pfs dh-group

Данной командой устанавливается номер группы метода Диффи-Хеллмана. Номер группы определяет уровень защищенности IPsec-соединения при обмене ключами – защищенность возрастает с ростом номера группы, но увеличивается и время установления соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
pfs dh-group <DH-GROUP>
no pfs dh-group
```

Параметры

<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14-31].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
rtt(config-isec-proposal)# pfs dh-group 5
```

pre-shared-key

Данной командой устанавливается общий секретный ключ для аутентификации, должен совпадать у обоих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

Синтаксис

```
pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-TEXT> } | hexadecimal {
<HEX> | encrypted <ENCRYPTED-HEX> } }
no pre-shared-key
```

Параметры

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате(0xYYYY...) или (YYYY...);

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Значение по умолчанию

none

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# pre-shared-key hexadecimal abc123
```

proposal

Данной командой устанавливается привязка профиля протокола IKE к политике.

Использование отрицательной формы команды (no) удаляет привязку профиля протокола IKE.

Синтаксис

```
[no] proposal <NAME>
```

Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# proposal ike_prop1
```

reauthentication disable

Данной командой возможно отключить процесс реаутентификации IKEv2-сессий.

Использование отрицательной формы команды (no) включает процесс реаутентификации.

Синтаксис

```
[no] reauthentication disable
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-POLICY

Пример

```
rtt(config-ike-policy)# reauthentication disable
```

remote address

Данной командой устанавливается IP-адрес удаленного шлюза IPsec-туннеля.

Использование отрицательной формы команды (no) удаляет IP-адрес удаленного шлюза.

Синтаксис

```
remote address { <ADDR> | any }
no remote address
```

Параметры

<ADDR> – IP-адрес удаленного шлюза.

any – ключ, позволяющий принимать запросы на установление IKE-сессии от любого IP-адреса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# remote address 192.168.1.2
```

remote id

Данной командой устанавливается ожидаемый идентификатор удаленного шлюза IPsec-туннеля, получаемый от удаленной стороны для идентификации в процессе согласования IKE-ключей.

Использование отрицательной формы команды (no) удаляет настройку ожидаемого идентификатора удаленного шлюза.

Синтаксис

```
remote id { any | dns <NAME> | ipv4 <ADDR> | keyid <KEY> }
```

```
no remote id
```

Параметры

any – ключ, обозначающий "любой" идентификатор;

<NAME> – полное доменное имя хоста (FQDN). Пример записи доменного имени – router.example.loc;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<KEY> – текстовая строка длиной до 255 символов.

Необходимый уровень привилегий

10

Значение по умолчанию

Отсутствует

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# remote id dns router.example.loc
```

remote network

Данной командой устанавливается IP-адрес подсети получателя, а также IP-протокол и порт или назначается динамический пул адресов для удалённых клиентов, использующих XAUTH. Трафик, удовлетворяющий заданным критериям, будет направлен в IPsec-туннель.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети отправителя.

Синтаксис

```
remote network { dynamic pool <POOL> | <ADDR/LEN> [ protocol { <TYPE> | <ID> } [
port <PORT> ] ] | any }
no remote network { dynamic pool |<ADDR/LEN> [ protocol { <TYPE> | <ID> } [ port
<PORT> ] ] | any }
```

Параметры

<POOL> – выделенный динамический пул адресов для клиентов XAUTH;

<ADDR/LEN> – IP-подсеть получателя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

any – ключ, указывающий на необходимость шифрования любого исходящего трафика.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# remote network 192.168.0.0/24 protocol tcp port 22
```

remote network dynamic client

Данной командой включается получение списка удаленных сетей от IPsec-VPN-сервера.

При использовании отрицательной формы команды (no) отключается получение списка удаленных сетей от IPsec-VPN-сервера.

Синтаксис

```
[no] remote network dynamic client
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GW

Пример

```
rtt(config-ike-gw)# remote network dynamic client
```

retransmit jitter

Данной командой устанавливается уровень случайного разброса периода ожидания ответа на сообщения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
retransmit jitter <VALUE>
no retransmit jitter
```

Параметры

<VALUE> – максимальный процент разброса значений, принимает значения [0..100].

Значение по умолчанию

0 %

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# retransmit jitter 50
```

retransmit limit

Данной командой устанавливается ограничение максимального периода времени ожидания ответа на сообщения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
retransmit limit <SEC>
no retransmit limit
```

Параметры

<SEC> – максимальный период времени ожидания ответа на сообщения принимает значения [15..300] секунд.

Значение по умолчанию

0 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# retransmit limit 180
```

retransmit timeout

Данной командой устанавливается базовый период времени ожидания ответа на сообщения.

Для первого отправленного сообщения период времени ожидания ответа будет равен базовому периоду, указанному в данной команде, а для последующих попыток интервал ожидания будет рассчитан по формуле:

"retransmit timeout" * 1.8 ^ (N-1),

где N – номер попытки, предел которых указан в команде *retransmit tries*.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
retransmit timeout <SEC>
no retransmit timeout
```

Параметры

<SEC> – базовый период времени ожидания ответа на сообщения принимает значения [1..30] секунд.

Значение по умолчанию

4 секунды

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# retransmit timeout 2
```

retransmit tries

Данной командой устанавливается количество попыток повторной отправки сообщений после наступления таймаута ожидания ответа.

Для первого отправленного сообщения период времени ожидания ответа будет равен базовому периоду, указанному в команде *retransmit timeout*, а для последующих попыток интервал ожидания будет рассчитан по формуле:

"retransmit timeout" * 1.8 ^ (N-1),

где N – номер попытки, предел которых указан в данной команде.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
retransmit tries <TRIES>
retransmit tries
```

Параметры

<TRIES> – количество попыток повторной отправки сообщений в случае наступления таймаута ожидания ответа принимает значения от 1 до 10.

Значение по умолчанию

5 попыток

Необходимый уровень привилегий

10

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# retransmit tries 3
```

security ike gateway

Данной командой осуществляется переход в командный режим конфигурирования шлюза IKE SECURITY IKE GATEWAY. Если шлюз IKE с указанным именем не существует в конфигурации, то он будет создан.

Параметры шлюза включают в себя VTI-интерфейс, в который будет направляться трафик, политика и версия протокола IKE, а также режим перенаправления трафика в туннель.

Использование отрицательной формы команды (no) удаляет шлюз протокола IKE.

Синтаксис

```
[no] security ike gateway <NAME>
```

Параметры

<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-шлюзы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike gateway ike_gw1
rtt(config-ike-gw)#
```

security ike keyring

Данной командой создается набор ключей аутентификации для протокола IKE, который включает в себя наборы соответствий адресов удаленных IPsec-клиентов и ключей PSK, которые будут использоваться при аутентификации.

Использование отрицательной формы команды (no) удаляет шлюз протокола IKE.

Синтаксис

```
[no] security ike keyring <NAME>
```

Параметры

<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все наборы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike keyring ike_keyring1
rtt(config-ike-gw)#
```

security ike policy

Данной командой создается политика IKE, которая включает в себя профили протокола IKE, общий секретный ключ для аутентификации и режим согласования первой фазы протокола IKE.

Использование отрицательной формы команды (no) удаляет указанную политику. Команда устанавливает режим командной строки SECURITY IKE POLICY.

Синтаксис

```
[no] security ike policy <NAME>
```

Параметры

<NAME> – имя политики IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-политики.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike policy ike_pol1
rtt(config-ike-policy)#
```

security ike proposal

Данной командой создается профиль протокола IKE (Internet Key Exchange), который включает в себя параметры алгоритмов шифрования и аутентификации метода Диффи-Хеллмана, которые будут использоваться при согласовании параметров IKE со встречной стороной VPN соединения при создании Security Association (SA). Кроме того, профиль задаёт предельное время действия SA. Использование отрицательной формы команды (no) удаляет заданный профиль.

Синтаксис

```
[no] security ike proposal <NAME>
```

Параметры

<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IKE-профили.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike proposal ike_prop1
rtt(config-ike-proposal)#
```

security ike session reauthentication

Данной командой задается режим повторной аутентификации ассоциации безопасности IKE в протоколе IKE версии 2.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security ike session reauthentication <MODE>
no security ike session reauthentication
```

Параметры

<MODE> – режим повторной аутентификации ассоциации безопасности IKE в протоколе IKE версии 2, принимает следующие значения:

- break-before-make – в процессе повторной аутентификации текущая ассоциация безопасности IKE совместно с ассоциациями безопасности IPsec будет разрушена до того, как будет установлена новая ассоциация безопасности IKE.
- make-before-break – в процессе повторной аутентификации текущая ассоциация безопасности IKE совместно с ассоциациями безопасности IPsec будет существовать и использоваться до полного завершения процесса установления новой ассоциации безопасности IKE.

Значение по умолчанию

break-before-make

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike session reauthentication make-before-break
```

security ike session uniqueids

Данной командой задается режим переподключения клиентов XAUTH с одним логином/паролем.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
security ike session uniqueids <MODE>
no security ike session uniqueids
```

Параметры

<MODE> – режим переподключения, принимает следующие значения:

- no – установленное подключение XAUTH будет удалено, если для нового подключения XAUTH инициатором соединения будет отправлено уведомление "INITIAL_CONTACT", будет назначен ранее использованный IP-адрес. В противном случае, установленное соединение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес.
- never – установленное подключение XAUTH будет удержано. Для нового подключения XAUTH будет назначен новый IP-адрес. Уведомление "INITIAL_CONTACT" будет в любом случае проигнорировано.
- replace – установленное подключение XAUTH будет удалено. Для нового подключения XAUTH будет использован ранее использованный IP-адрес.
- keep – установленное подключение XAUTH будет удержано. Новое подключение XAUTH будет отклонено.

Значение по умолчанию

never

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ike session uniqueids replace
```

show security ike

Команда используется для просмотра списка шлюзов, политик или профилей.

Синтаксис

```
show security ike { gateway | policy | proposal } [<NAME>]
```

Параметры

gateway – при указании команды «gateway» будет выведен список сконфигурированных шлюзов;

policy – при указании команды «policy» будет выведен список сконфигурированных политик;

proposal – при указании команды «proposal» будет выведен список сконфигурированных профилей;

<NAME> – имя. При указании определенного имени шлюза, политики, профиля будет выведена подробная информация.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ike proposal
   Proposal
   ~~~~~~~~
Name           Auth      Encryption         DH   Hash         Lifetime
------------   -------   ----------------   --   ----------   ----------
aaa            pre-sha   3des               1    sha1         3600
               red-key
 rtt# show security ike policy
   Policy
   ~~~~~~
Name                           Mode         Proposal
----------------------------   ----------   ----------------------------------
ike_pol1                       main         ike_prop1
 rtt# show security ike gateway ik_gw
Description:                --
IKE Policy:                 ike_pol1
IKE Version:                v1-only
Mode:                       route-based
Binding interface:          vti1
IKE Dead Peer Detection:
    Action:                 none
    Interval:               2
    Timeout:                30
```

user

Данной командой задается имя пользователя для аутентификации IKE-GETWAY.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

После выполнения данной команды маршрутизатор переходит в режим конфигурирования пароля пользователя (config-profile).

Синтаксис

```
[no] user <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ACCESS-PROFILE

Пример

```
rtt(config-access-profile)# user connecter963
```

version

Данной командой задаётся версия протокола IKE.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
version <VERSION>
no version
```

Параметры

<version> – версия IKE-протокола: v1-only или v2-only.

Значение по умолчанию

v1-only

Необходимый уровень привилегий

15

Командный режим

CONFIG-IKE-GATEWAY

Пример

```
rtt(config-ike-gw)# version v2-only
```

20.3.2. Управление VPN. Настройки IPsec

authentication algorithm

Данной командой устанавливается алгоритм аутентификации. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения: none, md5, sha1, sha2-256, sha2-384, sha2-512.

Значение по умолчанию

sha1

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
rtt(config-ipsec-proposal)# authentication algorithm md5
```

clear security ipsec vpn

Данной командой осуществляется сброс одного из текущих VPN-соединений.

Синтаксис

```
clear security ipsec vpn  [ vrf <VRF> ] <NAME>
```

Параметры

<NAME> – имя VPN, задаётся строкой до 31 символа;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear security ipsec vpn IPSEC_MAIN_OFFICE
```

description

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

CONFIG-IPSEC-PROPOSAL

CONFIG-IPSEC-POLICY

Пример

```
rtt(config-ipsec-vpn)# description "VPN to Moscow Office"
```

enable

Данной командой активируется IPsec VPN.

Использование отрицательной формы команды (no) деактивирует IPsec VPN.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# enable
```

encryption algorithm

Данной командой устанавливается алгоритм шифрования. Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Параметры

<ALGORITHM> – протокол шифрования, принимает значения: null, des, 3des, blowfis28, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
rtt(config-ipsec-proposal)# encryption algorithm blowfish128
```

ike dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ike dscp <DSCP>
no ike dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

63

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike dscp 40
```

ike establish-tunnel

Командой устанавливается режим активации VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ike establish-tunnel <MODE>
no Ike establish-tunnel
```

Параметры

<MODE> – режим активации VPN:

- by-request – соединение активируется встречной стороной;
- route – соединение активируется при появлении трафика, маршрутизируемого в туннель;
- immediate – туннель активируется автоматически после применения конфигурации. Если произойдет закрытие соединения от удаленной стороны VPN при использовании режима immediate, тогда установить соединение повторно будет возможно только после перезагрузки маршрутизатора или перезапуске VPN через CLI маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike establish-tunnel route
```

ike gateway

Данной командой осуществляется привязка IKE-шлюза к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в mode.

Синтаксис

```
ike gateway <NAME>
no ike gateway
```

Параметры

<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike gateway ike_gw1
```

ike idle-time

Данной командой устанавливается значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA.

Использование отрицательной формы команды (no) отключает данный таймер.

Синтаксис

```
ike idle-time <TIME>
no ike idle-time
```

Параметры

<TIME> – интервал в секундах, принимает значения [4..86400].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike idle-time 3600
```

ike rekey disable

Данной командой отключается пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт.

Использование отрицательной формы команды (no) включает пересогласование ключей.

Синтаксис

```
[no] ike rekey disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike rekey disable
```

ike rekey margin

Данной командой можно настроить начало пересогласования ключей IKE-соединения до истечения времени жизни.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
Ike rekey margin { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
no ike rekey margin { seconds | packets | kilobytes }
```

Параметры

<SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой lifetime seconds, см. lifetime). Принимает значения [4..86400].

<PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой lifetime packets, см. lifetime). Принимает значения [4..86400].

<KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой lifetime kilobytes, см. lifetime). Принимает значения [4..86400].

Значение по умолчанию

Пересогласование ключей до истечения времени – за 540 секунд.

Пересогласование ключей до истечения объема трафика и количества пакетов – отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike rekey margin seconds 1800
```

ike rekey randomization

Данной командой устанавливается уровень случайного разброса значений параметров margin seconds, margin packets, margin kilobytes.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ike rekey randomization <VALUE>
no ike rekey randomization
```

Параметры

<VALUE> – максимальный процент разброса значений, принимает значения [1..100].

Значение по умолчанию

100%

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike rekey randomization 10
```

ike ipsec-policy

Данная команда устанавливает привязку IPsec-политики к VPN. Данная команда актуальна, только если в VPN выбран режим согласования ключей «ike». Настройка режима согласования ключей описана в mode.

Синтаксис

```
ike ipsec-policy <NAME>
no ike ipsec-policy
```

Параметры

<NAME> – имя IPsec-политики, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
```

lifetime

Данной командой устанавливается время жизни IPsec-туннеля.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
lifetime { seconds <SEC> | packets <PACKETS> | kilobytes <KB> }
no lifetime { seconds | packets | kilobytes }
```

Параметры

<SEC> – период времени жизни IPsec-туннеля, по истечении происходит пересогласование. Принимает значения [1140..86400] секунд;

<PACKETS> – количество пакетов, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400];

<KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..4608000] секунд.

Значение по умолчанию

3600 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-POLICY

Пример

```
rtt(config-ipsec-proposal)# lifetime seconds 3600
```

manual authentication algorithm

Данной командой устанавливается алгоритм аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
manual authentication algorithm <ALGORITHM>
no manual authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации, принимает значения [md5, md5-128, sha1, sha1-160, aesxcbc, sha2-256, sha2-384, sha2-512].

Значение по умолчанию

none

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual authentication algorithm sha1
```

manual authentication key

Данной командой устанавливается ключ аутентификации. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Синтаксис

```
manual authentication key { ascii-text {<TEXT> | encrypted <ENCRYPTED-TEXT>} |
hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
no manual authentication key
```

Параметры

<TEXT> – строка [1..64] ASCII-символов;

<HEX> – число размером [1..32] байт, задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED_TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED_HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual authentication key hexadecimal abcdef
```

manual bind-interface vti

Данной командой указывается туннельный интерфейс, через который будет проходить трафик в режиме туннеля route-based. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
manual bind-interface vti <VTI>
no manual bind-interface vti
```

Параметры

<VTI> – индекс интерфейса VTI, принимает значения:

- R100/200 – [1..250];
- R800 – [1..500].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual bind-interface vti 0
```

manual encryption algorithm

Данной командой устанавливается алгоритм шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

```
manual encryption algorithm <ALGORITHM>
no manual encryption algorithm
```

Параметры

<ALGORITHM> – алгоритм шифрования, принимает значения: des, 3des, blowfis28, blowfis92, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.

Значение по умолчанию

3des

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual encryption algorithm blowfis28
```

manual encryption key

Данной командой устанавливается ключ шифрования. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) удаляет установленное значение.

Синтаксис

```
manual encryption key { ascii-text { < TEXT> | encrypted <ENCRYPTED-TEXT> } |
hexadecimal { <HEX> | encrypted <ENCRYPTED-HEX> } }
no manual encryption key
```

Параметры

<TEXT> – строка [1..36] ASCII-символов;

<HEX> – число размером [1..24] байт, задаётся строкой [2..72] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...);

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..24] байт, задаётся строкой [2..72] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..36] байт, задаётся строкой [2..144] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual encryption key hexadecimal 0x123456
```

manual mode

С помощью данной команды осуществляется установка режима перенаправления трафика в туннель. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
manual mode <MODE>
no manual mode
```

Параметры

<MODE> – режим прохождения трафика:

- policy-based – трафик перенаправляется на основе принадлежности к указанным в политиках подсетям;
- route-based – трафик перенаправляется на основе маршрутов, у которых шлюзом является туннельный интерфейс.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual mode route-based
```

manual protocol

Данной командой устанавливается инкапсулирующий протокол. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
manual protocol <TYPE>
no manual protocol
```

Параметры

<TYPE> – тип протокола, принимает значения:

- ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;
- esp – данный протокол осуществляет аутентификацию и шифрование трафика.

Значение по умолчанию

esp

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual protocol ah
```

manual spi

Данной командой устанавливается индекс параметров безопасности. Данная команда актуальна, только если в VPN выбран режим согласования ключей «manual». Настройка режима согласования ключей описана в mode.

Использование отрицательной формы команды (no) удаляет индекс параметров безопасности.

Синтаксис

```
manual spi <HEX>
no manual spi
```

Параметры

<HEX> – индекс параметров безопасности, задаётся значение размером 32 бита (8 символов) в шестнадцатеричном формате (0xYYYY…) или (YYYY…).

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# manual spi FF
```

mode

Данной командой устанавливается режим согласования данных, необходимых для активации VPN.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим работы VPN:

- ike – согласование алгоритмов аутентификации и шифрования, ключей аутентификации и шифрования, индекса параметра безопасности и других данных осуществляется через протокол IKE;
- manual – пользователь должен сам настроить идентичные параметры на обоих узлах для работы VPN. При данном режиме не происходит установления IKE-соединения между узлами. Каждый из узлов шифрует и дешифрует пакеты, основываясь только на заданных параметрах.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# mode ike
```

proposal

Данной командой к политике привязываются профили набора протоколов IPsec.

Использование отрицательной формы команды (no) удаляет привязку к указанному профилю.

Синтаксис

```
[no] proposal <NAME>
```

Параметры

<NAME> – имя профиля набора протоколов IPsec, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-POLICY

Пример

```
rtt(config-ipsec-policy)# proposal ipsec_prop1
```

protocol

Данной командой устанавливается инкапсулирующий протокол.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
protocol <PROTOCOL>
no protocol
```

Параметры

<PROTOCOL> – инкапсулирующий протокол, принимает значения:

- ah – данный протокол осуществляет только аутентификацию трафика, шифрование данных не выполняется;
- esp – данный протокол осуществляет аутентификацию и шифрование трафика.

Значение по умолчанию

esp

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-PROPOSAL

Пример

```
rtt(config-ipsec-proposal)# protocol ah
```

security ipsec policy

Данной командой создается политика набора протоколов IPsec, которая включает в себя профили набора протоколов IPsec для согласования второй фазы протокола IKE.

Использование отрицательной формы команды (no) удаляет установленное значение.

Команда устанавливает режим командной строки SECURITY IPSEC POLICY.

Синтаксис

```
[no] security ipsec policy <NAME>
```

Параметры

<NAME> – имя политики IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec-политики.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ipsec policy ipsec_pol1
rtt(config-ipsec-policy)#
```

security ipsec proposal

Данной командой создается профиль для набора протоколов IPsec. Профиль IPsec включает в себя параметры алгоритмов шифрования и аутентификации, протокола защиты соединения IPsec-туннеля, а также время жизни соединения.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Команда устанавливает режим командной строки SECURITY IPSEC PROPOSAL.

Синтаксис

```
[no] security ipsec proposal <NAME>
```

Параметры

<NAME> – имя профиля IPsec, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IPsec-профили.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ipsec proposal ipsec_prop1
rtt(config-ipsec-proposal)#
```

security ipsec vpn

Данной командой создается VPN на основе набора протоколов IPsec и устанавливается командный режим SECURITY IPSEC VPN.

Использование отрицательной формы команды (no) удаляет сконфигурированный VPN.

Синтаксис

```
[no] security ipsec vpn <NAME>
```

Параметры

<NAME> – имя VPN, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все VPN.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security ipsec vpn ipsec_vpn1
rtt(config-ipsec-vpn)#
```

show security ipsec

Данной командой выполняется просмотр конфигураций VPN, политик и профилей набора протоколов IPsec.

Синтаксис

```
show security ipsec { vpn configuration | policy | proposal } [<NAME>]
```

Параметры

vpn configuration – при указании данной команды будет выведена конфигурация всех VPN;

policy – при указании данной команды будет выведен список сконфигурированных политик набора протоколов IPsec;

proposal – при указании команды будет выведен список сконфигурированных профилей набора протоколов IPsec;

<NAME> – имя. При указании определенного имени VPN, политики или профиля будет выведена подробная информация.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ipsec proposal
   Proposal
   ~~~~~~~~
Name                    Prot   Enc. alg.          Auth. alg.        Lifetime
---------------------   ----   ----------------   ---------------   -----------
ipsec_prop1             esp    aes128             sha1              28800 sec
rtt# show secu   rity ipsec policy
Name                   Description           Proposal
--------------------   -------------------   ----------------------------------
ipsec_pol1                                   ipsec_prop1
Master# show security ipsec vpn configuration IPSECVPN
Description:                --
State:                      Enabled
IKE:
    Establish tunnel:           immediate
    IPsec policy:               IPSECPOLICY
    IKE gateway:                IKEGW
    IKE DSCP:                   63
    IKE idle-time:              0s
    IKE rekeying:               Enabled
        Margin time:                540s
        Margin kilobytes:           0
        Margin packets:             0
        Randomization:              100%
```

show security ipsec vpn authentication

Данная команда позволяет посмотреть список и параметры подключившихся IPsec-VPN-клиентов.

Синтаксис

```
show security ipsec vpn authentication <NAME> [ vrf <VRF> ]
```

Параметры

<NAME> – имя созданного IPsec VPN, задаётся строкой до 31 символа.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет включено разрешение DNS-имен.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ipsec vpn authentication
 Local host       Remote host      Local subnet         Remote subnet        Authentication                            State
 ---------------  ---------------  -------------------  -------------------  ----------------------------------------  -----------
 2.2.2.1          2.2.2.2          192.168.2.0/24       192.168.1.1/32       Xauth PSK, login: ipsec                   Established
```

show security ipsec vpn status

Данной командой выполняется просмотр статуса всех VPN, которые устанавливают соединение через IKE-протокол либо определенного VPN при указании его имени.

Синтаксис

```
show security ipsec vpn status [ vrf <VRF> ] [ <NAME> ]
```

Параметры

<NAME> – имя VPN, задаётся строкой до 31 символа;

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ipsec vpn status
 Name      Local host   Remote host  Initiator spi       Responder spi        State
 --------- ------------ ------------ ---------------     ---------------      ------
 ipsec_vpn1 10.100.14.1 10.100.14.2  0x05d8e0ac3543f0cb  0xcfa1c4179d001154   Established
```

type

Данной командой задается тип инкапсуляции IPsec.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
type <TYPE>
no type
```

Параметры

<TYPE> – тип инкапсуляции IPsec, принимает значения:

- tunnel;
- transport.

Значение по умолчанию

tunnel

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-access-vpn)# type transport
```

20.4. Управление VPN. Настройки удаленного доступа

20.4.1. Общие команды настройки удаленного доступа

authentication required disable

Данной командой отключается обязательная аутентификация PPP-клиентов для PPTP или L2TP-сервера.

Использование отрицательной формы команды (no) включается обязательная аутентификация PPP для PPTP- или L2TP-сервера.

Синтаксис

```
[no] authentication required disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-pptp-server)# authentication required disable
```

clear remote-access counters

Данной командой осуществляется сброс счетчиков соединений OpenVPN, PPTP, Wireguard, L2TP over IPsec-пользователей.

Синтаксис

```
clear remote-access counters [ pptp | l2tp | openvpn | wireguard ] [ server
<SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADRR> ]
```

Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP, Wireguard или L2TP over IPsec-сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec-пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec-пользователя.

При выполнении команды без параметра будут сброшены все счетчики соединений OpenVPN, PPTP, Wireguard и L2TP over IPsec-пользователей.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear remote-access counters
```

clear remote-access session

Данной командой осуществляется завершение соединений OpenVPN, PPTP и L2TP over IPsec-пользователей.

Синтаксис

```
clear remote-access session [ pptp | l2tp | openvpn] [ server <SERVER-NAME> ] [
username <USER-NAME> ] [ip-address <ADRR> ]
```

Параметры

<SERVER-NAME> – имя профиля OpenVPN, PPTP или L2TP over IPsec-сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec-пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec-пользователя. При выполнении команды без параметра будут завершены все OpenVPN, PPTP и L2TP over IPsec-соединения.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear remote-access session
```

description

Команда используется для изменения описания профиля OpenVPN, PPTP и L2TP over IPsec-серверов.

Использование отрицательной формы команды (no) удаляет описание профиля.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-SERVER-PEER

Пример

Установить описание для профиля PPTP-сервера:

```
rtt(config-pptp-server)# description "Our remote workers"
```

enable

Данной командой активируется конфигурируемый профиль серверов удаленного доступа.

Использование отрицательной формы команды (no) деактивирует конфигурируемый профиль.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-SERVER-PEER

Пример

```
rtt(config-pptp-server)# enable
```

encryption mppe

Данная команда включает шифрование MPPE (Microsoft Point-to-Point Encryption) для PPTP-соединений.

Использование отрицательной формы команды (no) отключает шифрование.

Синтаксис

```
[no] encryption mppe
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPTP-SERVER

Пример

```
rtt(config-pptp-server)# encryption mppe
```

remote-access

Данной командой создается профиль сервера удаленного доступа.

Использование отрицательной формы команды (no) удаляет указанный профиль.

Синтаксис

```
[no] remote-access <SERVER-TYPE> <NAME>
```

Параметры

<SERVER-TYPE> – тип сервера удаленного доступа. Может принимать значения l2tp, openvpn, pptp, wireguard.

<NAME> – имя профиля сервера удаленного доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# remote-access l2tp remote-workers
rtt(config-l2tp-server)#
```

show remote-access configuration

Командой выполняется просмотр параметров профилей OpenVPN, WireGuard PPTP и L2TP over IPsec-серверов.

Синтаксис

```
show remote-access configuration { pptp | l2tp | openvpn | wireguard} [ <NAME> ]
```

Параметры

<NAME> – имя профиля OpenVPN, WireGuard, PPTP или L2TP over IPsec сервера.

При выполнении команды без параметра будут показаны параметры всех OpenVPN, WireGuard, PPTP или L2TP over IPsec серверов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show remote-access configuration pptp pptp1
State:                      Enabled
Description:                --
Security zone:              trusted
Authentication mode:        local
MTU:                        1500
Local address:              192.168.1.1
Remote address:             rem_pptp(10.0.10.20-10.0.10.40)
Outside address:            115.0.0.1
DNS server:                 --
WINS server:                --
   Users
   ~~~~~
#     Name                    State      Encrypted password
---   --------------------    --------   ------------------------------
0     pptp                    Enabled    8CB5107EA7005AFF
1     petr                    Enabled    CCE5513EE45A1EAC
```

show remote-access counters

Командой выполняется просмотр счетчиков соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

Синтаксис

```
show remote-access counters [ pptp | l2tp | openvpn | wireguard] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADRR> ]
```

Параметры

<SERVER-NAME> – имя профиля WireGuard, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя;

При выполнении команды без параметра будут показаны счетчики всех соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show remote-access counters
User            IP-address        UC recv      Bytes recv   Err recv     MC recv
-------------   ---------------   ----------   ----------   ----------   ----------
ivan            10.20.20.5        262          25365        0            0
fedor           20.20.20.160      59           5236         0            0
User            IP-address        UC sent      Bytes sent   Err sent
-------------   ---------------   ----------   ----------   ----------
ivan            10.20.20.5        249          29298        0
fedor           20.20.20.160      16           739          0
rtt# show remote-access counters l2tp
PPTP Server: remote-workers
User: ivan(10.20.20.5)
 Packets received:               231
 Bytes received:                 22229
 Dropped on receive:             0
 Receive errors:                 0
 Multicasts received:            0
 Receive length errors:          0
 Receive buffer overflow errors: 0
 Receive CRC errors:             0
 Receive frame errors:           0
 Receive FIFO errors:            0
 Receive missed errors:          0
 Receive compressed:             0
 Packets transmitted:            189
 Bytes transmitted:              21858
 Dropped on transmit:            0
 Transmit errors:                0
 Transmit aborted errors:        0
 Transmit carrier errors:        0
 Transmit FIFO errors:           0
 Transmit heartbeat errors:      0
 Transmit window errors:         0
 Transmit comressed:             0
 Collisions:                     0
```

show remote-access status

Командой выполняется просмотр состояния соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

Синтаксис

```
show remote-access status [ pptp | l2tp | openvpn | wireguard] [ server <SERVER-NAME> ] [ username <USER-NAME> ] [ ip-address <ADRR> ]
```

Параметры

<SERVER-NAME> – имя профиля OpenVPN, WireGuard, PPTP или L2TP over IPsec сервера;

<USER-NAME> – имя OpenVPN, PPTP или L2TP over IPsec пользователя;

<ADDR> – IP-адрес OpenVPN, PPTP или L2TP over IPsec пользователя.

При выполнении команды без параметра будет показано состояние всех соединений OpenVPN, WireGuard, PPTP и L2TP over IPsec пользователей.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show remote-access status
User                IP-address         Server
-----------------   ---------------    --------------------------------
ivan                10.20.20.5         pptp(remote-workers)
fedor               20.20.20.160       l2tp(remote-workers-l2tp)
Count sessions: 2
```

20.4.2. Настройка L2TP over IPsec/PPTP-сервера

authentication method

Данной командой разрешает использование метода аутентификации удаленных пользователей, подключающихся по протоколам PPTP или L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

Синтаксис

```
[no] authentication method <METHOD>
```

Параметры

<METHOD> – метод аутентификации, принимает значения [chap, mschap, mschap-v2, eap, pap].

Значение по умолчанию

Разрешен только chap.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# authentication method mschap
```

authentication mode

Данной командой устанавливается режим аутентификации удаленных пользователей, подключающихся по протоколам PPTP или L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет установленный режим.

Синтаксис

```
authentication mode { local | radius }
no authentication mode
```

Параметры

- local – режим аутентификации, использующий локальную базу пользователей конфигурируемого профиля.
- radius – режим, при котором аутентификация пользователей проходит через RADIUS-сервер.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# authentication mode local
```

dns-servers

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

Синтаксис

```
dns-servers object-group <NAME>
no dns-servers
```

Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# dns-servers object-group pptp_dns
```

dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов PPTP и L2TP over IPsec-серверов.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
dscp <DSCP>
no dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

32

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# dscp 40
```

ipsec authentication method

Данной командой выбирается метод аутентификации по ключу для IKE-соединения. Аутентификация сообщений по ключу используется при установлении IKE-соединения, ключ задаётся командой *ipsec authentication pre-shared-key* (см раздел ipsec authentication pre-shared-key).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipsec authentication method pre-shared-key
no ipsec authentication method
```

Параметры

pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP-SERVER

Пример

```
rtt(config-l2tp-server)# ipsec authentication method pre-shared-key
```

ipsec authentication pre-shared-key

Данной командой устанавливается общий секретный ключ для аутентификации, который должен совпадать у обоих сторон, устанавливающих туннель.

Использование отрицательной формы команды (no) удаляет установленный ключ.

Синтаксис

```
ipsec authentication pre-shared-key { ascii-text { <TEXT> | encrypted <ENCRYPTED-
TEXT> }| hexadecimal {<HEX> | encrypted <ENCRYPTED-HEX> } }
no ipsec authentication pre-shared-key
```

Параметры

<TEXT> – строка [1..64] ASCII-символов.

<HEX> – число размером [1..32] байт задаётся строкой [2..128] символов в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<ENCRYPTED-TEXT> – зашифрованный пароль размером [1..32] байт, задаётся строкой [2..128] символов;

<ENCRYPTED-HEX> – зашифрованное число размером [2..64] байт, задаётся строкой [2..256] символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP-SERVER

Пример

```
rtt(config-l2tp-server)# ipsec authentication pre-shared-key ascii-text password
```

ipsec ike proposal

Данной командой для L2TP-сервера назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IKE.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IKE.

Синтаксис

```
[no] ipsec ike proposal <NAME>
```

Параметры

<NAME> – имя ранее созданного профиля протокола IKE, задаётся строкой до 31 символа.

Значение по умолчанию

Ограничение отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP-SERVER

Пример

```
rtt(config-l2tp-server)# ipsec ike proposal IKE_PROPOSAL
```

ipsec ike rekey enable

Данной командой включается пересогласование ключей до разрыва IKE-соединения.

Использование отрицательной формы команды (no) отключает пересогласование.

Синтаксис

[no] ipsec ike rekey enable

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP

Пример

```
rtt(config-l2tp-server)# ipsec ike rekey enable
```

ipsec proposal

Данной командой для L2TP-сервера назначается шаблон, ограничивающий используемые методы аутентификации и шифрования протокола IPsec.

Использование отрицательной формы команды (no) удаляет ограничения на использование методов аутентификации и шифрования протокола IPsec.

Синтаксис

```
[no] ipsec proposal <NAME>
```

Параметры

<NAME> – имя ранее созданного профиля IPsec, задаётся строкой до 31 символа.

Значение по умолчанию

Ограничение отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-L2TP-SERVER

Пример

```
rtt(config-l2tp-server)# ipsec proposal IPSEC_PROPOSAL
```

local-address

Данной командой указывается IP-адрес, используемый PPTP или L2TP over IPsec сервером в качестве локального IP-адреса туннеля.

Использование отрицательной формы команды (no) удаляет настроенный локальный IP-адрес туннеля.

Синтаксис

```
local-address { object-group <NAME> | ip-address <ADDR> }
no local-address
```

Параметры

<NAME> – имя профиля IP-адресов, который содержит локальный IP-адрес туннеля, задаётся строкой до 31 символа.

<ADDR> – локальный IP-адрес туннеля задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# local-address object-group pptp_local
```

mtu

Данной командой указывается MTU для интерфейсов, которые будут создаваться при подключении удаленных пользователей по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

```
mtu <MTU>
no mtu
```

Параметры

<MTU> – значение MTU, принимает значения в диапазоне [1280..1500].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# mtu 1400
```

outside-address

Данной командой указывается IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений.

Использование отрицательной формы команды (no) удаляет настроенный адрес для прослушивания.

Синтаксис

```
outside-address { object-group <NAME> | ip-address <ADDR> | interface { <IF> |
<TUN> } }
no outside-address
```

Параметры

<NAME> – имя профиля IP-адресов, содержащий адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся строкой до 31 символа.

<ADDR> – IP-адрес, который будет прослушиваться PPTP или L2TP over IPsec сервером на наличие входящих подключений, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# outside-address object-group pptp_outside
```

remote-address

Данной командой указывается список IP-адресов, из которого PPTP или L2TP over IPsec сервером выдаются динамические IP-адреса удаленным пользователям.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

Синтаксис

```
remote-address { object-group <NAME>| address-range <FROM-ADDR>-<TO-ADDR> }
no remote-address
```

Параметры

<NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа.

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# remote-address object-group pptp_remote
```

remote network

Данной командой в таблицу маршрутизации добавляется информация о IP-подсети, которая будет доступна со стороны клиента при установлении динамического PPTP/L2TP-тунеля.

Использование отрицательной формы команды (no) удаляет IP-адрес подсети, доступной через динамический PPTP/L2TP-туннель.

Синтаксис

```
remote network <ADDR/LEN>
no remote network
```

Параметры

<ADDR/LEN> – IP-подсеть получателя. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPP-USER

Пример

```
rtt(config-ppp-user)# remote network 192.168.54.0/24
```

remote networks

Данной командой в таблицу маршрутизации добавляется информация о IP-подсетях, которые будут доступны со стороны клиента при установлении динамического PPTP/L2TP-тунеля.

Использование отрицательной формы команды (no) удаляет информацию о IP-подсетях, доступных через динамический PPTP/L2TP-туннель.

Синтаксис

```
remote networks <OBJ-GROUP-NETWORK-NAME>
no remote network
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv4/IPv6-адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPP-USER

Пример

```
rtt(config-ppp-user)# remote network 192.168.54.0/24
```

username

Данной командой создается пользователь для подключения к PPTP или L2TP over IPsec серверам. После выполнения данной команды маршрутизатор переходит в режим конфигурирования параметров PPP-пользователя.

Использование отрицательной формы команды (no) удаляет указанного пользователя.

Команда устанавливает режим командной строки PPTP USER или L2TP USER в зависимости от текущего командного режима.

Синтаксис

```
[no] username <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# username fedor
rtt(config-pptp-user)#
```

wins-servers

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи, подключающиеся по протоколам PPTP и L2TP over IPsec.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

Синтаксис

```
wins-servers object-group <NAME>
no wins-servers
```

Параметры

<NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PPTP-SERVER

CONFIG-L2TP-SERVER

Пример

```
rtt(config-pptp-server)# wins-servers object-group l2tp_wins
```

20.4.3. Настройка OpenVPN-сервера

address-range

Данной командой указывается список IP-адресов, из которого OpenVPN-сервером выдаются динамические IP-адреса удаленным пользователям в режиме L2.

Использование отрицательной формы команды (no) удаляет список IP-адресов удаленных пользователей.

Синтаксис

```
address-range <FROM-ADDR>-<TO-ADDR>
no address-range
```

Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# address-range 192.168.1.10-192.168.1.250
```

authentication algorithm

Данная команда определяет алгоритм аутентификации клиентов OpenVPN.

Использование отрицательной формы команды (no) устанавливает режим аутентификации по умолчанию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication-algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- 8-128 bits key size: md4, rsa-md4, md5, rsa-md5, mdc2, rsa-mdc2;
- 8-160 bits key size: sha, sha1, rsa-sha, rsa-sha1, rsa-sha1-2, dsa, dsa-sha, dsa-sha1, dsa-sha1-old, ripemd160, rsa-ripemd160, ecdsa-with-sha1;
- 8-224 bits key size: sha-224, rsa-sha-224;
- 8-256 bits key size: sha-256, rsa-sha-256;
- 8-384 bits key size: sha-384, rsa-sha-384;
- 8-512 bits key size: sha-512, rsa-sha-512, whirlpool.

Значение по умолчанию

sha

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# authentication algorithm cleartext
```

bridge-group

Данная команда используется для включения клиентских соединений по OpenVPN в L2-домен.

Использование отрицательной формы команды (no) исключает соединения из L2-домена.

Синтаксис

```
bridge-group <BRIDGE-ID>
no bridge-group
```

Параметры

<BRIDGE-ID> – идентификационный номер моста. Задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# bridge-group 15
```

crypto

Данной командой указываются сертификаты и ключи. Сертификаты и ключи должны быть предварительно скопированы на маршрутизатор с помощью команды *copy*, описанной в разделе copy.

Использование отрицательной формы команды (no) удаляет из профиля указанный сертификат.

Синтаксис

```
crypto <CERTIFICATE-TYPE> <NAME>
no crypto <CERTIFICATE-TYPE>
```

Параметры

<CERTIFICATE-TYPE> – тип сертификата или ключа, может принимать следующие значения:

- ca – сертификат удостоверяющего сервера;
- cert – сертификат OpenVPN-сервера;
- crl – список отозванных сертификатов;
- dh – ключ Диффи-Хеллмана;
- private-key – приватный ключ сервера;
- ta – HMAC-ключ.

<NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# crypto ca ca.crt
```

client-isolation

Данной командой включается блокировка передачи данных между клиентами.

Использование отрицательной формы команды (no) снимает блокировку.

Синтаксис

```
[no] client-isolation
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# client-isolation
```

client-max

Данной командой устанавливается максимальное количество одновременных пользовательских сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
client-max <VALUE>
no client-max
```

Параметры

<VALUE> – максимальное количество пользователей, принимает значения [1..65535].

Значение по умолчанию

Не ограничено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# clients-max 500
```

compression

Данной командой включается механизм сжатия передаваемых данных между клиентами и сервером OpenVPN.

Использование отрицательной формы команды (no) отключает механизм сжатия передаваемых данных.

Синтаксис

```
[no] compression
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# compression
```

dns-server

Данной командой указывается список DNS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса DNS-серверов.

Синтаксис

```
dns-server <ADDR>
no dns-server { <ADDR> | all }
```

Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные диапазоны IP-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# dns-server 1.1.1.1
```

duplicate-cn

Данная команда разрешает подключение множества пользователей с одним сертификатом.

Использование отрицательной команды (no) запрещает использование одного сертификата более чем одному пользователю.

Синтаксис

```
[no] duplicate-cn
```

Параметры

Команда не имеет параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# duplicate-cn
```

encryption algorithm

Данной командой выбирается алгоритм шифрования, используемый при передачи данных.

Использование отрицательной формы команды (no) отключает шифрование.

Синтаксис

```
encryption algorithm <ALGORITHM>
no encryption algorithm
```

Параметры

<ALGORITHM> – идентификатор протокола шифрования, принимает значения: des, blowfish128, aes128, des-ede, aes192, 3des, desx, aes256.

Значение по умолчанию

Шифрование отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# encryption algorithm aes128
```

ip address

Данная команда определяет статический IP-адрес для указанного клиента. Использование отрицательной формы команды (no) удаляет статический IP-адрес у клиента.

Синтаксис

```
[no] ip address <ADDR>
```

Параметры

<ADDR> – IP-адрес, имеет следующий формат:

AAA.BBB.CCC.DDD – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-USER

Пример

```
rtt(config-openvpn-server)# username client
rtt(config-openvpn-user)# ip address 10.10.100.15
```

login authentication

Данной командой осуществляется активация списка аутентификации пользователей для их авторизации.

В конфигурации по умолчанию активным является список с именем «default», данный список содержит один способ аутентификации – «local».

Использование отрицательной формы команды (no) деактивирует список аутентификации.

Синтаксис

```
login authentication <NAME>
no login authentication
```

Параметры

<NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

default

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# login authentication OPENVPN-LIST
```

network

Данной командой определяется подсеть, из которой выдаются IP-адреса пользователям. Первый IP-адрес в подсети выступает шлюзом для пользовательских сессий.

Использование отрицательной формы команды (no) удаляет данную подсеть.

Синтаксис

```
network <ADDR/LEN>
no network
```

Параметры

<ADDR/LEN> – IP-подсеть подсети, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..29].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# network 192.168.25.0/24
```

port

Данной командой устанавливается TCP/UDP-порт, который будет прослушивается OpenVPN-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – TCP/UDP-порт, принимает значения [1..65535].

Значение по умолчанию

1194

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# port 5000
```

protocol

Данной командой устанавливается инкапсулирующий протокол.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
protocol <PROTOCOL>
no protocol
```

Параметры

<TYPE> – тип инкапсуляции, возможные значения:

- TCP-инкапсуляция в TCP-сегменты;
- UDP-инкапсуляция в UDP-дейтаграммы.

Значение по умолчанию

Остановлено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# protocol udp
```

redirect-gateway

Данная команда включает анонсирование маршрута по умолчанию для OpenVPN-соединений, что приводит к замене маршрута по умолчанию на клиентской стороне. Новым шлюзом по умолчанию станет IP-адрес OpenVPN-сервера.

Использование отрицательной формы команды (no) отключает анонсирование маршрута по умолчанию.

Синтаксис

```
[no] redirect-gateway
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# redirect-gateway
```

route

Данной командой включается анонсирование указанных подсетей, шлюзом является IP-адрес OpenVPN-сервера (первый IP-адрес из подсети, заданной с помощью команды network, описанной в разделе network).

Использование отрицательной формы команды (no) отключает анонсирование указанных подсетей.

Синтаксис

```
route <ADDR/LEN>
no route { <ADDR/LEN> | all }
```

Параметры

<ADDR/LEN> – IP-подсеть, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# route 192.168.25.0/24, 192.168.26.0/24
```

timers holdtime

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчёт от 0. Таймер сбрасывается при получении каждого ответа на keepalive-сообщение от встречной стороны. Рекомендуется устанавливать значение таймера не менее *3 * keepalive*.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers holdtime <TIME>
no timers holdtime
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# timers holdtime 360
```

timers keepalive

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers keepalive <TIME>
no timers keeaplive
```

Параметры

<TIME> – время в секундах, принимает значения [1..32767].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# timers keepalive 120
```

subnet

Данная команда определяет подсеть, которая должна быть доступна через подключение указанного пользователя OpenVPN-сервера. После подключения пользователя к OpenVPN-серверу в таблице маршрутизации появляется маршрут в указанную подсеть через динамически созданный туннель.

Использование отрицательной формы команды (no) удаляет подсеть, работающую за данным пользователем.

Синтаксис

```
[no] subnet <ADDRLEN>
```

Параметры

<ADDR/LEN> – IP-подсеть, имеет следующий формат:

AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [16..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-USER

Пример

```
rtt(config-openvpn-server)# username client
rtt(config-openvpn-user)# subnet 192.168.25.128/28
```

tunnel

Данной командой определяется тип соединения с частной сетью через OpenVPN-сервер.

Использование отрицательной формы команды (no) удаляет текущее значение.

Синтаксис

```
tunnel <TYPE>
no tunnel
```

Параметры

<TYPE> – инкапсулирующий протокол, принимает значения:

- ip – соединение точка-точка;
- ethernet – подключение к L2-домену.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# tunnel ip
```

username

Данная команда позволяет перейти в режим конфигурирования для указанного пользователя OpenVPN-сервера.

Использование отрицательной формы команды (no) восстанавливает настройки пользователя по умолчанию.

Синтаксис

```
[no] username { <NAME> | all }
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа;

all – ключ, используемый для удаления всех ранее созданных пользователей.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# username client
rtt(config-openvpn-user)#
```

wins-server

Данной командой указывается список WINS-серверов, которые будут использовать удаленные пользователи.

Использование отрицательной формы команды (no) удаляет настроенные адреса WINS-серверов.

Синтаксис

```
wins-server <ADDR>
no wins-server { <ADDR> | all }
```

Параметры

<ADDR> – IP-адрес WINS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

all – удалить все сконфигурированные IP-адреса DNS-серверов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OPENVPN-SERVER

Пример

```
rtt(config-openvpn-server)# wins-servers 1.1.1.1
```

20.4.4. Настройка WireGuard-сервера

access-addresses

Данной командой указывается список IP-адресов, которым будет разрешено находиться внутри туннеля.

Синтаксис

```
access-addresses { address-range <ADDR>[-<ADDR>] | prefix <ADDR/LEN> | object-group
<OBJ-GROUP-NETWORK-NAME> }
no access-addresses
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов, с которых разрешено принимать пакеты в этом peer. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255];

prefix <ADDR/LEN> – IP-подсеть, с которой разрешено принимать пакеты в этом peer. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32];

object-group <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER-PEER

Пример

```
rtt(config-wireguard-server-peer)# access-addresses object-group client
```

local-address

Данной командой устанавливается статический IP-адрес конфигурируемого сервера.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

Синтаксис

local-address <ADDR/LEN>

no local-address

Параметры

<ADDR/LEN> – IP-адрес и длина маски подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIREGUARD-SERVER

Пример

```
rtt(config-wireguard-server)# local-address 10.10.0.1/24
```

mtu

Данной командой указывается MTU для интерфейсов, которые будут создаваться при подключении удаленных пользователей.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

mtu <MTU>

no mtu

Параметры

<MTU> – 552–10000.

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIREGUARD-SERVER

Пример

```
rtt(config-wireguard-server)# mtu 1420
```

peer

Данная команда используется для перехода к настройкам разрешённых туннелей.

Использование отрицательной формы команды (no) удаляет разрешённый туннель.

Синтаксис

[no] peer <COUNT>

Параметры

<COUNT> – номер соответствующего пира, принимает значения [1..16].

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER

Пример

```
rtt(config-wireguard-server)# peer 1
```

port

Данной командой устанавливается UDP-порт, который будет прослушиваться WireGuard-сервером.

Синтаксис

port <PORT>

no port

Параметры

<PORT> – UDP-порт, принимает значения [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER

Пример

```
rtt(config-wireguard-server)# port 43021
```

pre-shared-key

Данной командой указывается необходимый симметричный ключ WireGuard.

Синтаксис

pre-shared-key <TYPE> <WORD>

no pre-shared-key

Параметры

<TYPE> – тип аргумента, устанавливаемого в качестве симметричного ключа:

- **ascii-text** – указать симметричный ключ в виде ASCII-текста, который будет сконвертирован в формат Base64;
- **base64** – указать симметричный ключ в формате Base64;
- **encrypted** – указать симметричный ключ в зашифрованном виде.

<WORD> – вводимый симметричный ключ. В зависимости от типа аргумента имеет длину 32 символа для ascii-text, 44 символа для формата base64 и 64 символа для зашифрованного вида.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER-PEER

Пример

```
rtt(config-wireguard-server-peer)# pre-shared-key base64
hHj3XWiLpb49G5yhKYF0bYKIMhExEtRnN+1sLbWwl8s=
```

private-key

Данной командой указывается необходимый приватный ключ WireGuard-сервера.

Синтаксис

private-key <NAME>

no private-key

Параметры

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER

Пример

```
rtt(config-wireguard-server)# private-key server.priv
```

public-key

Данной командой указывается необходимый публичный ключ клиента WireGuard.

Синтаксис

public-key <NAME>

no public-key

Параметры

<NAME> – имя приватного ключа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIREGUARD-SERVER-PEER

Пример

```
rtt(config-wireguard-server-peer)# public-key client.pub
```

21. МАРШРУТИЗАЦИЯ

21.1. Общие настройки маршрутизации

authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом (доступно только для режимов CONFIG-RIP и CONFIG-OSPF-VLINK);
- md5 – пароль хешируется по алгоритму md5.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

Пример

```
rtt(config-rip)# authentication algorithm cleartext
```

authentication key

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов.

<ENCRYPTED-TEXT> – зашифрованный пароль размером 8-16 байт (16-32 символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF-VLINK

Пример

```
rtt(config-bgp-af)# authentication key ascii-text 123456789
rtt(config-bgp-af)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

enable

Данной командой включается конфигурируемый протокол маршрутизации, область, виртуальное соединение, соседство.

Использование отрицательной формы команды (no) отключает RIP-протокол.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF

CONFIG-OSPF-AREA

CONFIG-OSPF-VLINK

CONFIG-IPV6-OSPF

CONFIG-IPV6-OSPF-AREA

CONFIG-IPV6-OSPF-VLINK

CONFIG-ISIS

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример 1

```
rtt(config-rip)# enable
```

Пример 2

```
rtt(config-ospf)# enable
```

Пример 3

```
 rtt(config-isis)# enable
```

Пример 4

```
rtt(config-bgp-neighbor)# enable
```

graceful-restart

Данной командой включается механизм поддержание активного состояния соседства с сохранением маршрутной информации на момент перезапуска протокола маршрутизации.

Использование отрицательной формы команды (no) отключает данный механизм.

Синтаксис

```
[no] graceful-restart
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF

CONFIG-IPV6-OSPF

Пример

```
rtt(config-bgp-neighbor)# graceful-restart
```

graceful-restart timeout

Данной командой устанавливается временной интервал, в течение которого поддерживать активное состояние соседства на момент перезапуска протокола маршрутизации. По истечению временного интервала встречная сторона считается недоступной. Используется совместно с командой graceful-restart.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
graceful-restart timeout <TIME>
no graceful-restart timeout
```

Параметры

<TIME> – время в секундах, принимает значения [1..1800].

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-OSPF

CONFIG-IPV6-OSPF

Пример

```
rtt(config-bgp-neighbor)# graceful-restart timeout 60
```

ip path-mtu-discovery age-timer

Данной командой настраивается временной интервал, в течение которого поддерживается текущий режим работы PMTU. По истечении данного времени режим PMTU сменится на тот, который указан в конфигурации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip path-mtu-discovery age-timer <TIME>
no ip path-mtu-discovery age-timer
```

Параметры

<TIME> – время в минутах, принимает значения [1..30];

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip path-mtu-discovery age-timer 15
```

ip path-mtu-discovery disable

Данная команда запрещает поиск PMTU для протоколов TCP, SCTP, DCCP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip path-mtu-discovery disable
no ip path-mtu-discovery disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Поиск PMTU разрешен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip path-mtu-discovery disable
```

ip path-mtu-discovery min-mtu

Данной командой настраивается минимальное значение MTU для использования в PMTU.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip path-mtu-discovery min-mtu <VALUE>
no ip path-mtu-discovery min-mtu
```

Параметры

<VALUE> – длина в байтах, принимает значения [64..10000].

Значение по умолчанию

552

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip path-mtu-discovery min-mtu 1000
```

ip path-mtu-discovery mode

Данной командой настраивается режим работы PMTU.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip path-mtu-discovery mode <MODE>
no ip path-mtu-discovery mode
```

Параметры

<MODE> – режим работы PMTU:

- default – режим работы по умолчанию, в соответствии RFC1191;
- icmp-discard – игнорировать входящие PMTU-сообщения;
- secure – выполнять фрагментацию только для протоколов TCP, SCTP and DCCP.

Значение по умолчанию

default

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip path-mtu-discovery mode icmp-discard
```

ip protocols max-routes

Данная команда позволяет настроить емкость таблиц маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip protocols <PROTOCOL> max-routes <VALUE>
no ip protocols <PROTOCOL> max-routes
```

Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, isis, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- BGP
  - R800 – [0..5000000];
  - R100/200 – [0..2500000].
- OSPF и IS-IS
  - R800 – [0..500000];
  - R100/200 – [0..300000].
- RIP
  - R100/200/800 – [0..10000];

Значение по умолчанию для глобального режима

BGP

- R800 – (5000000);
- R100/200 – (2500000).

OSPF и IS-IS

- R800 – (500000);
- R100/200 – (300000).

RIP

- R100/200/800 – (10000).

Значение по умолчанию для vrf

0

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-VRF

Пример

```
rtt(config)# ip protocols ospf max-routes 4400
```

ip protocols preference

Данная команда позволяет настроить приоритетность протоколов маршрутизации для основной таблицы маршрутизации.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip protocols <PROTOCOL> preference <VALUE>
no ip protocols <PROTOCOL> preference
```

Параметры

<PROTOCOL> – вид протокола, принимает значения: static, rip, ospf, isis, bgp, dhcp, l2tp, pppoe, pptp, nhrp;

<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255].

Значение по умолчанию

BGP (170)

IS-IS (160)

OSPF (150)

RIP (100)

DHCP (40)

L2TP (50)

PPPoE (50)

PPTP (50)

NHRP (20)

Static (1)

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip protocols ospf preference 44
```

ip tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

Синтаксис

```
ip tcp adjust-mss <MSS>
no ip tcp adjust-mss
```

Параметры

<MSS> – значение MSS, принимает значения в диапазоне [500..1460].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-IG

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip tcp adjust-mss 1400
```

ipv6 protocols max-routes

Данная команда позволяет настроить емкость таблиц маршрутизации IPv6.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 protocols <PROTOCOL> max-routes <VALUE>
no ipv6 protocols <PROTOCOL> max-routes
```

Параметры

<PROTOCOL> – вид протокола, принимает значения: rip (только в глобальном режиме), ospf, isis, bgp;

<VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне:

- BGP
  - R800 – [0..5000000];
  - R100/200 – [0..2500000].
- OSPFv3 и IS-IS
  - R800 – [0..500000];
  - R100/200 – [0..300000].
- IPv6 BGP
  - R800 – [0..5000000];
  - R100/200 – [0..1500000].

Значение по умолчанию для глобального режима

OSPFv3

- R800 – (500000);
- R100/200 – (300000).

IPv6 BGP

- R800 – (5000000);
- R-100/200 – [1..2500000].

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-VRF

Пример

```
rtt(config)# ipv6 protocols ospf max-routes 4400
```

ipv6 tcp adjust-mss

Данной командой переопределяется значение поля MSS (Maximum segment size) для IPv6 во входящих TCP-пакетах.

Использование отрицательной формы команды (no) отключает корректировку значение поля MSS.

Синтаксис

```
ipv6 tcp adjust-mss <MSS>
no ipv6 tcp adjust-mss
```

Параметры

<MSS> – значение MSS, принимает значения в диапазоне [40..1940].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-IG

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# ipv6 tcp adjust-mss 1400
```

show ip protocols

Данная команда выводит информацию о настройках протоколов IP-маршрутизации.

Синтаксис

```
show ip protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

Параметры

<PROTOCOL> – протокол маршрутизации, по которому должна быть отображена информация:

- bgp;
- ospf;
- isis;
- rip;
- static;
- dhcp;
- nhrp;
- pptp;
- pppoe;
- l2tp.

Без указания протокола маршрутизации, выводится информация о настройках всех протоколов маршрутизации.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh ip protocols
BGP:
    Max routes:     --
    Preference:     170
OSPF:
    Max routes:     --
    Preference:     150
RIP:
    Max routes:     --
    Preference:     100
Static:
    Preference:     1
```

show ip route

Данная команда позволяет просмотреть таблицу маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

Синтаксис

```
show ip route [ vrf <VRF> ] [ { <SUBNET> [ long-prefix ] | all | summary | arp-proxy | <PROTOCOL> } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];
- AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32];
- all – выводит информацию о всех маршрутах, включая не выбранные для FIB;
- long-prefix – выводит информацию о маршрутах до сетей являющихся подсетью заданной;
- summary – выводит суммарную статистику протоколов маршрутизации;
- arp-proxy – выводит информацию о настроенных пулах arp-nat-proxy на интерфейсах и туннелях;
- <PROTOCOL> – вывести все маршруты, полученные по определённому протоколу (bgp, connected, ospf , isis, rip, static, nhrp).

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip route
 Codes: C - connected, S - static, R - RIP derived,
        O - OSPF derived, IA - OSPF inter area route,
        E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
        B - BGP derived, D - DHCP derived, K - kernel route, V - VRRP route
        i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        H - NHRP, * - FIB route

C     * 192.168.1.0/24    [0/0]   dev br1                          [direct 01:14:16]
C     * 10.100.100.0/24   [0/0]   dev gi1/0/5                      [direct 01:14:17]
rtt# show ip route summary
Direct Connected: 12
Static:           46
RIP:              0
OSPF:             2000
BGP:              100000
DHCP:             1
IS-IS:            0
NHRP:             12
```

show ipv6 protocols

Данная команда выводит информацию о настройках протоколов IPv6-маршрутизации.

Синтаксис

```
show ipv6 protocols [ <PROTOCOL> ] [ vrf <VRF> ]
```

Параметры

<PROTOCOL> – протокол маршрутизации, по которому должна быть отображена информация:

- bgp;
- ospf;
- isis;
- static;
- dhcp;
- pppoe;
- pptp;
- l2tp.

Без указания протокола маршрутизации выводится информация о настройках всех протоколов маршрутизации.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh ipv6 protocols
BGP:
    Max routes:     --
    Preference:     170
OSPF:
    Max routes:     --
    Preference:     150
Static:
    Preference:     1
```

show ipv6 route

Команда для просмотра таблицы маршрутизации устройства. Если задан параметр <SUBNET>, то детально отображаются маршруты к данной подсети. Если задан параметр <VRF>, то команда отображает таблицу маршрутизации указанного экземпляра VRF.

Синтаксис

```
show ipv6 route [ vrf <VRF> ] [ { <SUBNET> | all | summary | <PROTOCOL> } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, опциональный параметр, может быть задан в следующих видах:

- X:X:X:X::X – IPv6-адрес хоста, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];
- X:X:X:X::X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];
- all – выводит информацию о всех маршрутах, включая неактивные;
- summary – выводит суммарную статистику протоколов маршрутизации;
- <PROTOCOL> – вывести все маршруты, полученные по определённому протоколу (bgp, connected, ospf , isis, rip, static).

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 route
 Codes: C - connected, S - static, R - RIP derived,
        O - OSPF derived, IA - OSPF inter area route,
        E1 - OSPF external type 1 route, E2 - OSPF external type 2 route
        B - BGP derived, D - DHCP derived, K - kernel route,
        * - FIB route
S     * ::/0              [1/0]   via fc00::1 on gi1/0/5           [static 03:16:23]
S     * 2001::/120        [1/6]   dev gi1/0/5                      [static 03:16:23]
C     * fc00::/120        [0/0]   dev gi1/0/5                      [direct 03:16:23]
S     * fc00:3::1/128     [1/0]   via fc00::1 on gi1/0/5           [static 03:16:23]
rtt# show ipv6 route summary
Direct Connected: 1
Static:           3
RIP:              0
OSPF:             0
BGP:              0
```

21.2. Общие команды анонсирования и приема маршрутов

default-information-originate

Данной командой включается генерация и отправка маршрута по умолчанию, если он прописан в таблице маршрутизации FIB, для:

- NSSA-области (анонсирование маршрута) в качестве NSSA-LSA;
- BGR IPv4/IPv6 address family.

Использование отрицательной формы команды (no) отключает генерацию маршрута по умолчанию.

Синтаксис

```
[no] default-information-originate
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

CONFIG-OSPF-AREA

CONFIG-IPV6-OSPF-AREA

Пример

```
rtt(config-ospf-area)# default-information-originate
```

description

Данная команда используется для изменения описания конфигурируемого списка IP-подсетей или IPv6-подсетей.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание списка IP-подсетей или IPv6-подсетей, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PL

CONFIG-IPV6-PL

Пример

```
rtt(config-route-map-rule)# description "Drop Local NETs"
```

ip prefix-list

Данной командой создается список IP-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

Синтаксис

```
[no] ip prefix-list <NAME>
```

Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip prefix-list ospf_in
```

ipv6 prefix-list

Данной командой создается список IPv6-подсетей, который в дальнейшем будет использоваться для фильтрации анонсируемых и получаемых IPv6-маршрутов.

Использование отрицательной формы команды (no) удаляет список префиксов.

Синтаксис

```
[no] ipv6 prefix-list <NAME>
```

Параметры

<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 prefix-list ospfv3_in
```

network

Данной командой включается анонсирование указанной подсети.

При использовании в CONFIG-OSPF-AREA и CONFIG-IPV6-OSPF-AREA подсеть анонсируется как External Type 2 (LSA5).

Использование отрицательной формы команды (no) отключает анонсирование указанной подсети.

Синтаксис

```
[no] network <ADDR/LEN> [route-map <NAME>]
```

Параметры

<ADDR/LEN> – IP-подсеть, имеет один из следующих форматов:

- AAA.BBB.CCC.DDD/EE – IP-адрес подсети с маской в форме префикса, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];
- X:X:X:X::X/EE – IPv6-адрес и маска подсети, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа;

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-RIPNG

CONFIG-OSPF-AREA

CONFIG-IPV6-OSPF-AREA

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

Пример

```
rtt(config-bgp)# network 192.168.25.0/24
```

permit/deny

Данной командой разрешаются (permit) или запрещаются (deny) списки префиксов.

Синтаксис

```
permit [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN> | <IPV6-ADDR/LEN> } ]
[ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
deny [ { object-group <OBJ-GROUP-NETWORK-NAME> | <ADDR/LEN > | <IPV6-ADDR/LEN> } ]
[ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] } ]
no { object-group <OBJ-GROUP-NETWORK-NAME> | prefix { <ADDR/LEN> | <IPV6-ADDR/LEN>
} }
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP/IPv6-адресов, задаётся строкой до 31 символа;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов и [1..128] в IPv6-списках префиксов;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PL

CONFIG-IPV6-PL

Пример

```
rtt(config-pl)# permit static ge 24 le 28
```

prefix-list

Данной командой добавляется фильтрация подсетей во входящих или исходящих обновлениях.

Использование отрицательной формы команды (no) отключает фильтрацию.

Синтаксис

```
prefix-list <PREFIX-LIST-NAME> { in | out }
no prefix-list { in| out }
```

Параметры

<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа:

in – фильтрация входящих маршрутов;

out – фильтрация анонсируемых маршрутов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-rip)# prefix-list rip_in in
```

redistribute bgp

Данной командой включается анонсирование маршрутов автономной системы BGP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов автономной системы BGP.

Синтаксис

```
redistribute bgp <AS> [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ unit <ID> ] [
<LEVEL> ]
no redistribute bgp <AS>
```

Параметры

<AS> – номер автономной системы из которой будут анонсироваться маршруты, может принимать значения [1..4294967295];

<TYPE> – тип атрибута OSPF Metric, принимает значения type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа;

<ID> – номер юнита, принимает значения [1..4].

<LEVEL> — уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-bgp-af)# redistribute bgp 30
rtt(config-ospf)# redistribute bgp 35
rtt(config-rip)# redistribute bgp 300
```

redistribute connected

Данной командой включается анонсирование напрямую подключенных подсетей.

Использование отрицательной формы команды (no) отключает анонсирование напрямую подключенных подсетей.

Синтаксис

```
redistribute connected [metric <TYPE> <METRIC>][ route-map <NAME> ] [ <LEVEL> ]
no redistribute connected
```

Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF).

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.

<LEVEL> — уровень протокола IS-IS в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-rip)# redistribute connected
```

redistribute ipv6 bgp

Данной командой включается анонсирование IPv6-маршрутов автономной системы BGP.

Использование отрицательной формы команды (no) отключает анонсирование IPv6-маршрутов автономной системы BGP.

Синтаксис

```
redistribute ipv6 bgp <AS> [ route-map <NAME> ] [ <LEVEL> ]
no redistribute ipv6 bgp <AS>
```

Параметры

<AS> – номер автономной системы из которой будут анонсироваться IPv6-маршруты, может принимать значения [1..4294967295];

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых BGP-IPv6-маршрутов, задаётся строкой до 31 символа.

<LEVEL> — уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне;

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# redistribute ipv6 bgp 30 level 2
```

redistribute ipv6 ospf

Данной командой включается анонсирование маршрутов из базы OSPFv3-процесса согласно выбранным условиям.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы OSPFv3-процесса.

Синтаксис

```
redistribute ipv6 ospf <ID> [ <ROUTE-TYPE1> ] [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ] [ <ROUTE-TYPE4> ] [ route-map <NAME> ]
no redistribute ipv6 ospf <ID>
```

Параметры

<ID> – номер процесса, может принимать значение [1..65535].

<ROUTE-TYPE> – типы маршрутов которые будут анонсироваться:

- intra-area – анонсирование маршрутов OSPFv3-процесса в пределах зоны;
- inter-area – анонсирование маршрутов OSPFv3-процесса между зонами;
- external1 – анонсирование внешних маршрутов OSPFv3-формата 1;
- external2 – анонсирование внешних маршрутов OSPFv3-формата 2.

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых OSPFv3-маршрутов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# redistribute ipv6 ospf 10 external2
```

redistribute isis

Данной командой включается анонсирование маршрутов из базы ISIS согласно выбранным условиям. Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы ISIS.

При редистрибьюции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

Синтаксис

```
redistribute isis <ID> <ROUTE-TYPE> [metric <TYPE> <METRIC>] [ route-map <NAME> ]
no redistribute isis <ID>
```

Параметры

<ID> – номер процесса, может принимать значение [1..65535].

<ROUTE-TYPE> – тип маршрута:

- level-1 – анонсирование маршрутов ISIS-процесса уровня 1;
- level-2 – анонсирование маршрутов ISIS-процесса уровня 2;
- inter-area – анонсирование межзоновых маршрутов IS-IS-процесса.

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых IS-IS-маршрутов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-bgp-af)# redistribute isis 1 level-1 level-2
```

redistribute ospf

Данной командой включается анонсирование маршрутов из базы OSPF-процесса согласно выбранным условиям.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы OSPF-процесса.

При редистрибьюции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

Синтаксис

```
redistribute ospf <ID> <ROUTE-TYPE1> [ <ROUTE-TYPE2> ] [ <ROUTE-TYPE3> ] [ <ROUTE-TYPE4> ] [metric <TYPE> <METRIC>] [ route-map <NAME> ]
no redistribute ospf <ID>
```

Параметры

<ID> – номер процесса, может принимать значение [1..65535];

<ROUTE-TYPE> – типы маршрутов которые будут анонсироваться:

- intra-area – анонсирование маршрутов OSPF-процесса в пределах зоны;
- inter-area – анонсирование маршрутов OSPF-процесса между зонами;
- external1 – анонсирование внешних маршрутов OSPF-формата 1;
- external2 – анонсирование внешних маршрутов OSPF-формата 2.

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-bgp-af)# redistribute ospf 10 external2
```

redistribute rip

Данной командой включается анонсирование маршрутов из базы маршрутов RIP.

Использование отрицательной формы команды (no) отключает анонсирование маршрутов из базы RIP.

При редистрибьюции маршрутов из базы OSPF-процесса iBGP-пиру по умолчанию BGP next-hop атрибут изменен не будет.

Синтаксис

```
redistribute rip [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ <LEVEL> ]
no redistribute rip
```

Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.

<LEVEL> — уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

Пример

```
rtt(config-bgp)# redistribute rip
```

redistribute static

Данной командой включается анонсирование статических IPv4/IPv6-маршрутов.

Использование отрицательной формы команды (no) отключает анонсирование статических маршрутов.

Синтаксис

```
redistribute static [metric <TYPE> <METRIC>] [ route-map <NAME> ] [ <LEVEL> ]
no redistribute static
```

Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2 (актуально только в режиме конфигурирования CONFIG-OSPF):

- type-1 – устанавливает тип метрики E1 для анонсируемого маршрута;
- type-2 – устанавливает тип метрики E2 для анонсируемого маршрута.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535] (актуально только в режиме конфигурирования CONFIG-OSPF);

<NAME> – имя маршрутной карты, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа;

<LEVEL> – уровень протокола IS-IS, в который происходит анонсирование маршрутов (актуально только в режиме конфигурирования CONFIG-ISIS):

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне;

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-bgp)# redistribute static
```

21.3. Маршрутизация на основе политик (PBR)

action

Данная команда используется для указания действия, которое должно быть применено для маршрутной информации, удовлетворяющей заданным критериям. Используется только для фильтрации маршрутов протоколов динамической маршрутизации и не имеет действия при конфигурировании PBR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
action <ACT>
no action
```

Параметры

<ACT> – назначаемое действие:

- permit – прием/анонсирование маршрутной информации разрешено;
- deny – прием/анонсирование маршрутной информации запрещено.

Значение по умолчанию

permit

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action deny
```

action add community

Данной командой задается значение атрибута BGP Community, которое будет добавлено в существующий список.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action add community { no-advertise | no-export | <COMMUNITY-LIST> [ track <TRACK-ID> default <COMMUNITY-LIST> ]}
no action add community
```

Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N-часть принимает значения [0..65535]. Можно указать до 64 community;

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не

является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

```
rtt(config-route-map-rule)# action add community no-advertise
```

action add extcommunity

Данной командой задается значение атрибута BGP ExtCommunity, которое будет установлено в существующий список.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action add extcommunity <EXTCOMMUNITY-LIST> [ track <TRACK-ID> default
<EXTCOMMUNITY-LIST> ]
no action add extcommunity
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

```
rtt(config-route-map-rule)# action add extcommunity rd:65500:100
```

action remove community

Данной командой удаляется значение атрибута BGP Community в существующем списке.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action remove community { no-advertise | no-export | <COMMUNITY-LIST> [ track
<TRACK-ID> default <COMMUNITY-LIST> ]}
no action remove community
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N-часть принимает значения [0..65535]. Можно указать до 64 community;

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

```
rtt(config-route-map-rule)# action remove community 65500:100
```

action remove extcommunity

Данной командой удаляется значение атрибута BGP ExtCommunity в существующем списке.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action remove extcommunity <EXTCOMMUNITY-LIST> [ track <TRACK-ID> default
<EXTCOMMUNITY-LIST> ]
no action remove extcommunity
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

```
rtt(config-route-map-rule)# action remove extcommunity rt:65500:1
```

action set as-path prepend

Данной командой устанавливается значение атрибута BGP AS-Path, которое будет добавляться в начало списка автономных систем в маршруте. При конфигурировании правила, управляемого средствами Tracking, также имеется возможность установить значение по умолчанию после ключевого слова default. Это значение будет установлено в случае если условия объекта отслеживания не будут выполнены.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set as-path prepend <AS-PATH> [ track <TRACK-ID> default <AS-PATH> ]
no action set as-path prepend
```

Параметры

<AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем;

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set as-path prepend 100,200,300
```

action set as-path replace

Данной командой включается опция замены номера или последовательности номеров AS в атрибуте BGP AS-Path маршрута на локальный номер AS.

Использование отрицательной формы команды (no) отменяет замену.

Синтаксис

```
action set as-path replace { any | <AS-PATH> }
no action set as-path replace
```

Параметры

any – заменять любой номер автономной системы;

<AS-PATH> – список номеров автономных систем, который будет заменён на локальный номер AS. Задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set as-path prepend 100,200,300
```

action set community

Данной командой задается значение атрибута BGP Community, которое будет установлено в новый список (существующий список будет перезаписан).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set community {no-advertise | no-export | <COMMUNITY-LIST> [ track <TRACK-
ID> default <COMMUNITY-LIST> ]}
no action set community
```

Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:N,AS:N, где AS-часть принимает значения [0..65535], N часть принимает значения [0..65535]. Можно указать до 64 community;

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

no-advertise – при указании команды маршруты, которые передаются с данным значением атрибута community, не должны анонсироваться другим BGP-соседям;

no-export – при указании команды маршруты, которые передаются с таким значением атрибута community, не должны анонсироваться за пределы конфедерации (автономная система, которая не является частью конфедерации, считается конфедерацией). То есть, маршруты не анонсируются eBGP-соседям, но анонсируются внешним соседям в конфедерации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set community no-advertise
```

action set extcommunity

Данной командой задается значение атрибута BGP ExtCommunity, которое будет установлено в новый список (существующий список будет перезаписан).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set extcommunity <EXTCOMMUNITY-LIST> [ track <TRACK-ID> default
<EXTCOMMUNITY-LIST> ]
no action set extcommunity
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

<EXTCOMMUNITY-LIST> – список community, задаётся в виде KIND:AS:N,KIND:AS:N,KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) и ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 ExtCommunity.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set extcommunity ro:435:6
```

action set ip bgp-next-hop

Данной командой задается значение атрибута BGP Next-Hop, которое будет установлено в маршруте при получении или анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set ip bgp-next-hop <ADDR>
no action set ip next-hop
```

Параметры

<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set ip bgp-next-hop 10.100.100.1
```

action set ip next-hop

Данной командой задается значение Next-Hop, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set ip next-hop { <NEXTHOP> | blackhole | unreachable | prohibit }
no action set ip next-hop
```

Параметры

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

- blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;
- unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);
- prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13).

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set ip next-hop prohibit
```

action set ip next-hop verify-availability

Данной командой задается Next-Hop для пакетов, которые попадают под критерии в указанном списке доступа (ACL).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set ip next-hop verify-availability <NEXTHOP> <METRIC> [ track <TRACK-ID> ]
no action set ip next-hop verify-availability { <NEXTHOP> | all }
```

Параметры

<NEXTHOP> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

[METRIC] – метрика маршрута, принимает значения [1..255];

<TRACK-ID> – идентификатор tracking объекта, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Команда *action set ip next-hop с ключом verify-availability* применяется для создания route-map, которые будут использоваться для Policy-Based-Routing, а не для изменения next-hop в маршрутах, полученных по протоколу динамической маршрутизации.

Ключ `verify-availability` указывает, что данный `action` будет работать, только если активен IP-интерфейс маршрутизатора терминирующий подсеть к которой относится указанный в команде `<NEXTHOP>`.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set ip next-hop verify-availability 1.1.1.1 25
```

action set ipv6 bgp-next-hop

Данной командой задается значение атрибута BGP Next-Hop для IPv6, которое будет установлено в маршруте при анонсировании по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set ipv6 bgp-next-hop <ADDR>
no action set ipv6 next-hop
```

Параметры

<IPV6-ADDR> – IPv6-адрес шлюза, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set ipv6 bgp-next-hop 2002::765:1
```

action set ipv6 next-hop

Данной командой задается значение Next-Hop для IPv6, которое будет установлено в маршруте, полученном по BGP.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set ipv6 next-hop <NEXTHOP>
no action set ipv6 next-hop
```

Параметры

<NEXTHOP> – IPv6-адрес шлюза, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set ipv6 next-hop 55::205:2
```

action set local-preference

Данной командой задается значение атрибута BGP Local Preference, которое будет установлено в маршруте, а при использовании параметра increment/decrement – дельта, на которую будет увеличен или уменьшен этот атрибут относительно исходного значения соответственно.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set local-preference {<PREFERENCE> | increment < VALUE > | decrement < VALUE
>} [ track <TRACK-ID> default <PREFERENCE> ]}
no action set local-preference
```

Параметры

<VALUE> – значение дельты изменения атрибута BGP Local Preference относительно исходного значения. Принимает значение [1..2147483647]. Если в результате применения операции increment/decrement значение метрики выйдет за допустимый диапазон, значение Local Preference принимается равным максимально или минимально допустимому значению соответственно. ;

<PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [1..2147483647];

<TRACK-ID> – идентификатор tracking объекта, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set local-preference 120
```

action set metric bgp

Данной командой задается значение атрибута BGP MED, которое будет установлено в маршруте, а при использовании параметра increment/decrement – дельта, на которую будет увеличен или уменьшен этот атрибут относительно исходного значения соответственно. При конфигурировании правила, управляемого средствами Tracking, также имеется возможность установить значение по умолчанию после ключевого слова default. Это значение будет установлено в случае, если условия объекта отслеживания не будут выполнены.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set metric bgp {< METRIC > | increment < VALUE > | decrement < VALUE >} [
track <TRACK-ID> [ default <METRIC> ] ]
no action set metric bgp
```

Параметры

<VALUE> – значение дельты изменения атрибута BGP MED относительно исходного значения. Принимает значение [0..4294967295]. Если в результате применения операции increment/decrement значение метрики выйдет за допустимый диапазон, значение BGP MED принимается равным максимально или минимально допустимому значению соответственно;

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295];

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set metric bgp 10
```

action set metric isis

Данной командой задается значение атрибута IS-IS Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set metric isis <METRIC>
no action set metric isis
```

Параметры

<METRIC> – значение атрибута IS-IS Metric, принимает значения [1..16777215].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set metric isis 10
```

action set metric ospf

Данной командой задается значение атрибута OSPF Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set metric ospf <TYPE> <METRIC>
no action set metric ospf
```

Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set metric ospf type-1 10
```

action set metric rip

Данной командой задается значение атрибута RIP Metric, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set metric rip <METRIC>
no action set metric rip
```

Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set metric rip 5
```

action set origin

Данной командой задается значение атрибута BGP Origin, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set origin <ORIGIN> [ track <TRACK-ID> [ default <ORIGIN> ] ]
no action set origin
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100];

<ORIGIN> – значение атрибута BGP Origin, принимает следующие значения:

- egp – маршрут выучен по протоколу Exterior Gateway Protocol (EGP);
- igp – маршрут получен внутри исходной автономной системы;
- incomplete – маршрут выучен другим образом.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set origin igp
```

action set tag ospf

Данной командой задается значение атрибута OSPF Tag, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set tag ospf <TAG>
no action set tag ospf
```

Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set tag ospf 20
```

action set tag rip

Данной командой задается значение атрибута RIP Tag, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set tag rip <RIP>
no action set tag rip
```

Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set tag rip 20
```

action set weight bgp

Данной командой задается значение атрибута BGP weight, которое будет установлено в маршруте.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
action set weight bgp {< WEIGHT > | increment < VALUE > | decrement < VALUE >} [
track <TRACK-ID> [ default <WEIGHT> ] ]
no action set weight bgp
```

Параметры

<WEIGHT> – значение атрибута BGP weight, принимает значения [0..65535];

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# action set weight bgp 200
```

description

Данная команда используется для изменения описания конфигурируемого правила маршрутной карты.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание правила маршрутной карты, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# description "Drop Local NETs"
```

ip local policy route-map

Данной командой назначается политика маршрутизации на основе списков доступа (ACL). Данная политика маршрутизации работает для пакетов, которые генерирует сам маршрутизатор.

Использование отрицательной формы команды (no) удаляет политику маршрутизации.

Синтаксис

```
ip local policy [vrf <VRF>] route-map <NAME>
no local ip policy [vrf <VRF>] route-map
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра политика маршрутизации будет работать только в определенном VRF;

<NAME> – имя сконфигурированной политики маршрутизации, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip local policy route-map drop-local-net
```

ip policy route-map

Данной командой на интерфейс назначается политика маршрутизации на основе списков доступа (ACL). Данная политика маршрутизации работает для пакетов, приходящих через интерфейс, на котором она назначена.

Использование отрицательной формы команды (no) удаляет политику маршрутизации.

Синтаксис

```
ip policy route-map <NAME>
no ip policy route-map
```

Параметры

<NAME> – имя сконфигурированной политики маршрутизации, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

Пример

```
rtt(config-if-gi)# ip policy route-map drop-local-net in
```

match as-path

Данной командой устанавливается значение атрибута BGP AS-Path в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match as-path { [ begin | contain | end ] <AS-PATH> | empty | regex <REGEX> }
no match as-path
```

Параметры

<AS-PATH> – список номеров автономных систем, задаётся в виде AS,AS,AS, где каждая часть принимает значения [1..4294967295]. Можно указать до 10 номеров автономных систем.

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- begin – значение атрибута начинается с указанного списка номеров автономных систем;
- end – значение атрибута оканчивается на указанный список номеров автономных систем;
- contain – значение атрибута содержит указанный список номеров автономных систем;
- empty – значение атрибута пусто;
- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match as-path begin 100,200,300
```

match community

Данной командой устанавливается значение атрибута BGP Community в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match community { <COMMUNITY-LIST> | regex <REGEX> }
no match community
```

Параметры

<COMMUNITY-LIST> – список community, задаётся в виде AS:N,AS:,AS, где AS часть принимает значения [0..65535], N часть принимает значения [0..65535]. Можно указать до 64 community.

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match community 100:1,200:3,300:65000
```

match extcommunity

Данной командой устанавливается значение атрибута BGP ExtCommunity в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match extcommunity { <EXTCOMMUNITY-LIST> | regex <REGEX> }
no match extcommunity
```

Параметры

<EXTCOMMUNITY-LIST> – список extcommunity, задаётся в виде KIND:AS:N, KIND:AS:N, KIND:AS:N, где:

- KIND – тип extcommunity, принимает значения rt (Route Target) или ro (Route Origin);
- AS – номер автономной системы, принимает значения [1..4294967295];
- N – номер extcommunity, определяющий политику маршрутизации трафика, принимает значения [1..65535].

Можно указать до 64 extcommunity.

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match extcommunity ro:435:6
```

match ip access-group

Данной командой устанавливается ACL-группа, для которой должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip access-group <NAME>
no match ip access-group
```

Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip access-group ACCESS
```

match ip address

Данной командой устанавливается адреса маршрутов, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip address <ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ] }
no match ip address prefix
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<LEN> – длина префикса, принимает значения [1..32];

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip address 192.168.0.0/32 ge 16
```

match ip address object-group

Данной командой устанавливается профиль IP-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN> |
ge <LEN> [ le <LEN> ]
no match ip address object-group
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip address object-group local_nets
```

Данной командой устанавливается профиль IP-адресов, содержащий значения Next-Hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ip next-hop
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

match ip bgp next-hop

Данной командой устанавливается профиль IP-адресов, содержащий значения BGP атрибута Next-Hop, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ip bgp next-hop
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip bgp next-hop object-group Next-hop
```

match ip next-hop

Данной командой устанавливается профиль IP-адресов, содержащий значения next-hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ip next-hop
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов шлюзов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip next-hop object-group block_nexthop
```

match ip route-source

Командой устанавливается профиль IP-адресов. Профиль содержит IP-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IP-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip route-source object-group <OBJ-GROUP-NETWORK-NAME>
no match ip route-source
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который должен содержать диапазоны IP-адресов источника маршрутной информации, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ip route-source object-group source_routers
```

match ipv6 address

Данной командой устанавливается адреса IPv6-маршрутов, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ipv6 address <IPV6-ADDR/LEN> [ { eq <LEN> | le <LEN> | ge <LEN> [ le <LEN> ]
}
no match ipv6 address prefix
```

Параметры

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<LEN> – длина префикса, принимает значения [1..128];

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ipv6 address FF02:0:0:0:0:1:FF00::/104
```

match ipv6 address object-group

Данной командой устанавливается профиль IPv6-адресов, содержащий значения подсетей назначения в маршруте, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ipv6 address object-group <OBJ-GROUP- NETWORK -NAME> [ { eq <LEN> | le <LEN>
| ge <LEN> [ le <LEN> ]
no match ipv6 address object-group
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать префиксы подсетей назначения, задаётся строкой до 31 символа;

eq – при указании команды длина префикса должна соответствовать указанной;

le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;

ge – при указании команды длина префикса должна быть больше либо соответствовать указанной.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ipv6 address object-group local_nets
```

match ipv6 bgp next-hop

Данной командой устанавливается профиль IPv6-адресов, содержащий значения BGP атрибута Next-Hop, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ipv6 bgp next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ipv6 next-hop
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов шлюзов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ipv6 bgp next-hop object-group block_next-hop
```

match ipv6 next-hop

Данной командой устанавливается профиль IPv6-адресов, содержащий значения Next-Hop в таблице маршрутизации (show ip route) маршрутизатора, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ipv6 next-hop object-group <OBJ-GROUP-NETWORK-NAME>
no match ipv6 next-hop
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов шлюзов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ipv6 next-hop object-group block_nexthop
```

match ipv6 route-source

Командой устанавливается профиль IPv6-адресов. Профиль содержит IPv6-адреса маршрутизатора, анонсировавшего маршрут. Используется для фильтрации по IPv6-адресу источника при анонсировании маршрутной информации.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ipv6 route-source object-group <OBJ-GROUP-NETWORK-NAME>
no match ipv6 route-source
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IPv6-адресов, который должен содержать диапазоны IPv6-адресов источника маршрутной информации, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match ipv6 route-source object-group source_routers
```

match local-preference

Данной командой устанавливается значение атрибута BGP LOCAL_PREF в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match local-preference { <METRIC> | regex <REGEX> }
no match local-preference
```

Параметры

<METRIC> – значение атрибута BGP LOCAL_PREF, принимает значения [0..2147483647].

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

```
rtt(config-route-map-rule)# match local-preference 101
```

match metric bgp

Данной командой устанавливается значение атрибута BGP MED в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match metric bgp { <METRIC> | regex <REGEX> }
no match metric bgp
```

Параметры

<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match metric bgp 10
```

match metric ospf

Данной командой устанавливается значение атрибута OSPF Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match metric ospf <TYPE> <METRIC>
no match metric ospf
```

Параметры

<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2.

<METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match metric ospf type-1 10
```

match metric rip

Данной командой устанавливается значение атрибута RIP Metric в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match metric rip <METRIC>
no match metric rip
```

Параметры

<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match metric rip 5
```

match tag ospf

Данной командой устанавливается значение атрибута OSPF Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match tag ospf <TAG>
no match tag ospf
```

Параметры

<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match tag ospf 20
```

match tag rip

Данной командой устанавливается значение атрибута RIP Tag в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match set tag rip <RIP>
no match set tag rip
```

Параметры

<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match tag rip 20
```

match weight bgp

Данной командой устанавливается значение атрибута BGP weight в маршруте, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match weight bgp { <WEIGHT> | regex <REGEX> }
no match weight bgp
```

Параметры

<WEIGHT> – значение атрибута BGP weight, принимает значения [0..65535].

Опциональные параметры, с помощью которых задаётся частичное соответствие атрибута:

- regex – значение атрибута соответствует регулярному выражению, где <REGEX> – регулярное выражение, задаётся по стандарту POSIX-Extended Regular Expressions.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP-RULE

Пример

```
rtt(config-route-map-rule)# match bgp weight 20
```

route-map

Командой добавляется фильтрация и модификация маршрутов во входящих или исходящих направлениях.

Использование отрицательной формы команды (no) отключает фильтрацию и модификацию маршрутов в соответствующем направлении.

Синтаксис

```
route-map <NAME> <DIRECTION>
no route-map <DIRECTION>
```

Параметры

<NAME> – имя сконфигурированной маршрутной карты, задаётся строкой до 31 символа;

<DIRECTION> – направление:

- in – фильтрация и модификация получаемых маршрутов;
- out – фильтрация и модификация анонсируемых маршрутов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

CONFIG-OSPF

CONFIG-IPV6-OSPF

CONFIG-ISIS

CONFIG-RIP

CONFIG-RIPND

Пример

```
rtt(config-bgp-neighbor)# route-map drop-local-net in
```

route-map

Данной командой создается маршрутная карта, которая в дальнейшем будет использоваться для фильтрации и модификации анонсируемых и получаемых IP-маршрутов, и осуществляется переход в режим настройки параметров маршрутной карты.

Использование отрицательной формы команды (no) удаляет маршрутную карту.

Синтаксис

```
[no] route-map <NAME>
```

Параметры

<NAME> – имя конфигурируемой маршрутной карты, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# route-map drop-local-net
rtt(config-route-map)#
```

rule

Данной командой создается правило маршрутной карты с определённым номером и осуществляется переход в режим настройки параметров правила. Правила обрабатываются устройством в порядке возрастания номеров правил.

Использование отрицательной формы команды (no) удаляет правило по номеру либо все правила.

Критерии отбора, описанные в правиле, подчиняются логическому "И".

Например:

```
RTT(config)# route-map TEST
RTT(config-route-map)# rule 1
RTT(config-route-map-rule)# match community 65:100
RTT(config-route-map-rule)# match weight bgp 200
RTT(config-route-map-rule)# exit
```

Под правило попадут BGP маршруты, в атрибутах которых community и MED находятся значения 65:100 и 200 соответственно.

Правила обрабатываются устройством в порядке возрастания номеров правил. Если маршрут подпадает под все условия (match), описанные в правиле, выполняется настроенное в правиле действие (action) и дальнейший просмотр правил не производится.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1 .. 10000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ROUTE-MAP

Пример

```
rtt(config-route-map)# rule 2
rtt(config-route-map-rule)#
```

show ip route-map

Данная команда используется для просмотра маршрутных карт.

Синтаксис

```
show ip route-map <NAME> [ <ORDER> ]
```

Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip route-map drop-local-net
Order:                                  2
Description:                            Drop route to private nets
Matching pattern:
    Access group                            --
    AS path                                 --
    Community                               --
    Extcommunity                            --
    BGP local-preference:                   --
    BGP metric (MED):                       --
    BGP weight:                             --
    Address (object-group):                 local_net
    Next hop (object-group ipv4):           --
    Route source (object-group ipv4):       --
    Address (object-group ipv6):            --
    Next hop (object-group ipv6):           --
    Route source (object-group ipv6):       --
    IP address:                             --
    IPv6 address:                           --
    RIP metric                              --
    RIP tag                                 --
    OSPF metric type                        --
    OSPF metric                             --
    OSPF tag                                --
Actions:
    Decision:                               Deny
    Route next hop:                         --
    Route IPv6 next hop:                    --
    AS path (prepand):                      --
    Community (set):                        --
    Community (add):                        --
    Community (remove):                     --
    Extcommunity (set):                     --
    Extcommunity (add):                     --
    Extcommunity (remove):                  --
    Local preference:                       --
    BGP next hop address:                   --
    BGP IPv6 next hop address:              --
    BGP metric (MED):                       --
    BGP weight:                             --
    Origin:                                 --
    RIP metric                              --
    RIP tag                                 --
    OSPF metric type                        --
    OSPF metric                             --
    OSPF tag                                --
```

21.4. Настройка связок ключей

accept-lifetime

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации принятых пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
accept-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E>
<YEAR_E>
no accept-lifetime
```

Параметры

<TIME_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

<DAY_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH_B> – месяц начала использования ключа, принимает значения:

- January;
- February;
- March;
- April;
- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December.

<YEAR_B> – год начала использования ключа, принимает значения [2001..2037].

<TIME_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

<DAY_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH_E> – месяц окончания использования ключа, принимает значения:

- January;
- February;
- March;
- April;
- May;
- June;
- July;
- August;
- September;
- October;
- November;
- December.

<YEAR_E> – год окончания использования ключа, принимает значения [2001..2037].

Значение по умолчанию

Ключ действителен постоянно.

Необходимый уровень привилегий

15

Командный режим

CONFIG-KEYCHAIN-KEY

Пример

```
rtt(config-keychain-key)# accept-lifetime 16:35:00 10 May 2015 16:35:00 10 June
2021
```

key

Данной командой добавляется ключ в связку ключей и осуществляется переход в режим настройки параметров ключа.

Использование отрицательной формы команды (no) удаляет указанный ключ.

Синтаксис

```
[no] key <ID>
```

Параметры

<ID> – идентификатор ключа, задается в диапазоне [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-KEYCHAIN

Пример

```
rtt(config-keychain)# key 25
```

key-chain

Командой добавляется связка ключей в систему и осуществляется переход в режим настройки параметров связки ключей.

Использование отрицательной формы команды (no) удаляет указанный список.

Синтаксис

```
[no] key-chain <KEYCHAIN>
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, строка до 16 ASCII-символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# key-chain lock
```

key-string

Данной командой устанавливается пароль для аутентификации.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no key-string
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;
<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-KEYCHAIN-KEY

Пример

```
rtt(config-keychain-key)# key-string ascii-text 123456789
rtt(config-keychain-key)# key-string ascii-text encrypted CDE65039E5591FA3F1
```

send-lifetime

Данная команда определяет период времени, в течение которого данный ключ может использоваться для аутентификации при отправке пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
send-lifetime <TIME_B> <DAY_B> <MONTH_B> <YEAR_B> <TIME_E> <DAY_E> <MONTH_E>
<YEAR_E>
no send-lifetime
```

Параметры

<TIME_B> – устанавливаемое время начала использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

<DAY_B> – день месяца начала использования ключа, принимает значения [1..31];

<MONTH_B> – месяц начала использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR_B> – год начала использования ключа, принимает значения [2001..2037];

<TIME_E> – устанавливаемое время окончания использования ключа, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];

- SS – секунды, принимает значение [0 .. 59].

<DAY_E> – день месяца окончания использования ключа, принимает значения [1..31];

<MONTH_E> – месяц окончания использования ключа, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<YEAR_E> – год окончания использования ключа, принимает значения [2001..2037].

Значение по умолчанию

Ключ действителен постоянно.

Необходимый уровень привилегий

15

Командный режим

CONFIG-KEYCHAIN-KEY

Пример

```
rtt(config-keychain-key)# send-lifetime 16:35:00 15 May 2014 16:35:00 21 June 2018
```

21.5. Настройка параметров протокола BFD

clear bfd

Данная команда перезапускает процесс BFD. В результате выполнения команды все установленные BFD-сессии (IPv4/IPv6 в GRT/VRF) будут сброшены.

Синтаксис

```
clear bfd
```

Параметры

Не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear bfd
2022-01-04T21:00:19+00:00 %BFD-W-NEIG: Session to 192.168.39.230 Down
```

ip bfd idle-tx-interval

Данной командой задаётся интервал, по истечении которого происходит отправка BFD-сообщения соседу. Данный параметр служит для уменьшения числа генерируемых сообщений для тех случаев, когда BFD-сосед недоступен или на нем выключен протокол BFD.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip bfd idle-tx-interval <TIMEOUT>
no ip bfd idle-tx-interval
```

Параметры

<TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне:

- [200..65535] для R800;
- [300..65535] для R100/200.

Значение по умолчанию

1 секунда

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config)# ip bfd idle-tx-interval 4000
```

ip bfd log-adjacency-changes

Данной командой включается логирование изменений состояния BFD-протокола.

Использование отрицательной формы команды (no) отключает логирование изменений состояния BFD-протокола.

Синтаксис

```
ip bfd log-adjacency-changes
no ip bfd log-adjacency-changes
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip bfd log-adjacency-changes
```

ip bfd min-rx-interval

Данной командой задаётся минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. Данный параметр анонсируется соседу.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip bfd min-rx-interval <TIMEOUT>
no ip bfd min-rx-interval
```

Параметры

<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне:

- [200..65535] для R800;
- [300..65535] для R100/200.

Значение по умолчанию

- 200 миллисекунд на R800;
- 300 миллисекунд на R100/200.

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config)# ip bfd min-rx-interval 1000
```

ip bfd min-tx-interval

Данной командой задаётся минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. Данный параметр используется только когда BFD-сессия активна, в иных случаях используется ip bfd idle-tx-interval.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip bfd min-tx-interval <TIMEOUT>
no ip bfd min-tx-interval
```

Параметры

<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседу, принимает значение в миллисекундах в диапазоне:

- [200..65535] для R800;
- [300..65535] для R100/200.

Значение по умолчанию

- 200 миллисекунд на R800;
- 300 миллисекунд на R100/200.

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config)# ip bfd min-tx-interval 1000
```

ip bfd multiplier

Данной командой задаётся число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. Время обнаружения недоступности в каждом из направлений рассчитывается из данного числа, умноженного на tx/rx-интервал.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip bfd multiplier <COUNT>
no ip bfd multiplier
```

Параметры

<COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config)# ip bfd multiplier 10
```

ipv6 bfd passive

Данной командой BFD-сессия переводится в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip bfd passive
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Активный режим.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

Пример

```
rtt(config-if-gi)# ip bfd passive
```

ipv6 bfd log-adjacency-changes

Данной командой включается логирование изменений состояния IPv6 BFD-протокола.

Использование отрицательной формы команды (no) отключает логирование изменений состояния IPv6 BFD-протокола.

Синтаксис

```
ipv6 bfd log-adjacency-changes
no ipv6 bfd log-adjacency-changes
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Логирование выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 bfd log-adjacency-changes
```

show ip bfd

Данная команда отображает информацию о параметрах BFD-протокола или определенных интерфейсов при использовании фильтра.

Синтаксис

```
show ip bfd [ { interface <IF> | tunnel <TUN> } ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip bfd
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval:    1000 ms
Multiplier:          5 packets
Passive:             No
rtt# show ip bfd interface gi1/0/1
Minimum RX interval: 200 ms
Minimum TX interval: 200 ms
Idle TX interval:    1000 ms
Multiplier:          10 packets
Passive:             Yes
```

show bfd neighbors

Данная команда отображает список всех сессий, установленных (настроенных) по протоколу BFD.

Синтаксис

```
show bfd neighbords [ vrf <NAME_VRF> ] [ {<IPv4-ADDRESS> | <IPv6-ADDRESS> | client
{ bfd | static | bgp | ospf | isis | rip } } ]
```

Параметры

<NAME-VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<IPv4-ADDRESS> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPv6-ADDRESS> – IPv6-адрес соседа, задается в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

bfd – выводит информацию о настроенных (с помощью команды ip/ipv6 bfd neighbors) BFD-сессиях;

static – при указании команды отображаются BFD-сессии созданные для статических маршрутов;

bgp – выводит информацию о BFD-сессиях для протокола BGP.

ospf – выводит информацию о BFD-сессиях для протокола OSPF;

isis – выводит информацию о BFD-сессиях для протокола ISIS;

rip – выводит информацию о BFD-сессиях для протокола RIP;

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show bfd neighbors 192.168.39.230
Neighbor address:                   192.168.39.230
Local address:                      192.168.39.215
Interface:                          gi1/0/3
Remote discriminator:               3427539848
Local discriminator:                3379089541
State:                              Up
Session type:                       Control
Session mode:                       Single-hop
Local diagnostic code:              No Diagnostic
Remote diagnostic code:             No Diagnostic
Minimal Tx Interval:                300 ms
Minimal Rx Interval:                300 ms
Multiplier:                         5
Actual Tx Interval:                 1000 ms
Actual Detection Interval:          5000 ms
Number of transmitted packets:      497
Number of received packets:         338
Uptime (d,h:m:s):                   00,03:54:29
Client:                             BGP
Last received packet:
    Desired Min Tx Interval:        1000 ms
    Required Min Rx Interval:       1000 ms
    Multiplier:                     5
```

21.6. Настройка статических маршрутов IPv4/IPv6

gateway

Данной командой задаётся адрес, интерфейс или туннель, который будет выступать в качестве шлюза.

Использование отрицательной формы команды (no) удаляет шлюз с определённым адресом, интерфейсом или туннелем.

Синтаксис

```
gateway { <NEXTHOP> [ <WEIGHT> ] [ bfd ] [ track <TRACK-ID> ] [ unit <ID> ] | <IF>
[ <WEIGHT> ] | <TUN> [ <WEIGHT> ] }
no gateway { <NEXTHOP> [ unit <ID> ] | <IF> | <TUN> }
```

Параметры

<NEXTHOP> – адрес шлюза, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес шлюза, где каждая часть принимает значения [0..255];
- X:X:X:X::X – IPv6-адрес шлюза, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<WEIGHT> – метрика шлюза, принимает значения [0..255];

<ID> – номер юнита, принимает значения [1..4];

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<TRACK-ID> – идентификатор Tracking-объекта. Если шлюз привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте. Доступно только для ipv4.

bfd – при указании данного ключа активируется проверка доступности шлюза при помощи протокола BFD. В случае недоступности, шлюз удаляется из таблицы маршрутизации.

Необходимый уровень привилегий

10

Командный режим

CONFIG-MULTIPATH-ROUTE

CONFIG-MULTIPATH-IPV6-ROUTE

Пример

```
rtt(config-multipath-route)# gateway 192.168.0.1
```

ip bfd neighbor

Данной командой запускается работа механизма BFD с определенным IP-адресом. Работа такого механизма необходима для удаления статических маршрутов из таблицы маршрутизации при недоступности next-hop.

Использование отрицательной формы команды (no) останавливает работа механизма BFD с определенным IP-адресом.

Синтаксис

```
ip bfd neighbor <ADDR> [ { interface <IF> | tunnel <TUN> } ] [ local-address <ADDR>
[ multihop ] ] [ vrf <VRF> ]
no ip bfd neighbor <ADDR> [vrf <VRF>]
```

Параметры

<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip bfd neighbor 192.168.0.2
```

ip route

Команда позволяет создать статический IP-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

Синтаксис

```
ip route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [ bfd ] [ unit <ID> ] |
interface <IF> | tunnel <TUN> | blackhole | unreachable | prohibit } [ track
<TRACK-ID> ] [ name <NAME>] } | wan load-balance rule <RULE> } [ <METRIC> ]
no ip route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];
- AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32].

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

resolve – при указании данного параметра IP-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут будет отмечен как "Unreachable". Глубина поиска равна 5.

<ID> – номер юнита, принимает значения [1..4].

<IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

<RULE> – номер правила wan, задаётся в диапазоне [1..50].

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;

unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);

prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13);

<METRIC> – метрика маршрута, принимает значения [0..255].

<TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте.

<NAME> – имя (описание) маршрута, текстовая переменная длинной до 31 символа.

bfd – при указании данного ключа активируется удаление статического маршрута в случае недоступности next-hop. Для работы данного механизма должен быть запущен механизм BFD с IP-адресом next-hop (см. команду *ip bfd neighbor*).

Проверка next-hop при помощи протокола bfd. В случае недоступности next-hop маршрут удаляется.

Если в качестве подсети указать 0.0.0.0/0, то будет задан маршрут по умолчанию.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через шлюз 192.165.56.65:

```
rtt(config)# ip route 192.165.3.0/24 192.165.56.65 6
```

Пример 2

Задать маршрут до подсети 192.165.3.0/24 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
rtt(config)# ip route 192.165.3.0/24 interface gigabitethernet 1/0/5 6
```

Пример 3

Задать маршрут до подсети 192.165.3.0/24 через туннельный интерфейс vti 1:

```
rtt(config)# ip route 192.165.3.0/24 interface vti 1
```

ip route multipath

Команда позволяет создать статический multipath IP-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

Синтаксис

```
ip route multipath [ vrf <VRF> ] <SUBNET> [ track <TRACK-ID> ] [ name <NAME>] [
<METRIC> ]
no ip route multipath [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- AAA.BBB.CCC.DDD – IP-адрес хоста, где каждая часть принимает значения [0..255];
- AAA.BBB.CCC.DDD/NN – IP-адрес подсети с маской в виде префикса, где AAA-DDD принимают значения [0..255] и NN принимает значения [1..32].

<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<METRIC> – метрика маршрута, принимает значения [0..255];

<TRACK-ID> – идентификатор Tracking-объекта. Если маршрут привязан к Tracking-объекту, то он появится в системе только при выполнении всех условий, заданных в объекте;

<NAME> – имя (описание) маршрута, текстовая переменная длинной до 31 символа.

Если в качестве подсети указать 0.0.0.0/0, то будет задан маршрут по умолчанию.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip route multipath 192.165.3.0/24
```

ipv6 bfd neighbor

Данной командой запускается работа механизма BFD с определенным IPv6-адресом. Работа такого механизма необходима для удаления bgp-маршрутов из таблицы маршрутизации при недоступности next-hop.

Использование отрицательной формы команды (no) останавливает работа механизма BFD с определенным IPv6-адресом.

Синтаксис

```
ipv6 bfd neighbor <IPV6-ADDR> [ interface <IF> ] [ local-address <IPV6-ADDR> [
multihop ] ] [ vrf <VRF> ]
no ipv6 bfd neighbor <IPV6-ADDR> [ vrf <VRF> ]
```

Параметры

<IPV6-ADDR> IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 bfd neighbor FF02:0:0:0:0:1:FF00::/104
```

ipv6 route

Данная команда позволяет создать статический IPv6-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

Синтаксис

```
ipv6 route [ vrf <VRF> ] <SUBNET> { { <NEXTHOP> [ resolve ] [bfd] [ unit <ID> ] |
interface <IF> | blackhole | unreachable | prohibit [ <METRIC> ] [ name <NAME>] } |
wan load-balance rule <RULE> [ <METRIC> ] }
no ipv6 route [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- X:X:X:X::X – IPv6-адрес хоста, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
- X:X:X:X::X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

<NEXTHOP> – IPv6-адрес шлюза, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

resolve – при указании данного параметра IPv6-адрес шлюза будет рекурсивно вычислен через таблицу маршрутизации. Если при рекурсивном вычислении не удастся найти шлюз из напрямую подключенной подсети, то данный маршрут не будет установлен в систему.

<ID> – номер юнита, принимает значения [1..4].

<IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю.

unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1).

prohibit – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication administratively prohibited, code 13).

[METRIC] – метрика маршрута, принимает значения [0..255].

<NAME> – имя (описание) маршрута, текстовая переменная длинной до 31 символа.

bfd – при указании данного ключа, активируется проверка next-hop при помощи протокола bfd. В случае недоступности next-hop маршрут удаляется.

Если в качестве подсети указать ::/0, то будет задан маршрут по умолчанию.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример 1

Задать маршрут до подсети 2001::/120 с метрикой 6 через шлюз fc00::1:

```
rtt(config)# ipv6 route 2001::/120 fc00::1 6
```

Пример 2

Задать маршрут до подсети 2001::/120 с метрикой 6 через интерфейс GigabitEthernet 1/0/5:

```
rtt(config)# ipv6 route 2001::/120 interface gigabitethernet 1/0/5  6
```

Пример 3

Задать маршрут до подсети 2001::/120 через туннельный интерфейс vti 1:

```
rtt(config)# ipv6 route 2001::/120 interface vti 1
```

ipv6 route multipath

Данная команда позволяет создать статический multipath IPv6-маршрут к указанной подсети.

Использование отрицательной формы команды (no) удаляет указанный маршрут.

Синтаксис

```
ipv6 route multipath [ vrf <VRF> ] <SUBNET> [ name <NAME>] [ <METRIC> ]
no ipv6 route multipath [ vrf <VRF> ] <SUBNET> [ <METRIC> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<SUBNET> – адрес назначения, может быть задан в следующих видах:

- X:X:X:X::X – IPv6-адрес хоста, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];
- X:X:X:X::X/EE – IPv6-адрес подсети с маской в виде префикса, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

[METRIC] – метрика маршрута, принимает значения [0..255];

<NAME> – имя (описание) маршрута, текстовая переменная длинной до 31 символа;

bfd – при указании данного ключа активируется проверка доступности шлюза при помощи протокола BFD. В случае недоступности, шлюз удаляется из таблицы маршрутизации.

Если в качестве подсети указать ::/0, то будет задан маршрут по умолчанию.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 multipath route 2001::/120
```

21.7. Настройка протокола BGP

address-family

Данной командой осуществляется переход в режим настройки параметров передачи информации протоколов для процесса BGP.

Использование отрицательной формы команды (no) удаляет настроенные параметры передачи информации протоколов.

Синтаксис

```
[no] address-family { ipv4 | ipv6 } unicast
```

Параметры

- ipv4 – семейство ipv4;
- ipv6 – семейство ipv6.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# address-family ipv4
```

address-family

Данной командой определяется тип конфигурируемой маршрутной информации и переход в данный режим настройки.

Использование отрицательной формы команды (no) удаляет настроенные параметры передачи информации протоколов.

Синтаксис

```
[no] address-family { ipv4 | ipv6 | vpnv4 | l2vpn vpls } unicast
```

Параметры

- ipv4 – семейство ipv4;
- ipv6 – семейство ipv6;
- vpnv4 – семейство vpnv4;
- l2vpn vpls – семейство l2vpn.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# address-family ipv4
```

advertise-map

Данной командой активируется функция Conditional advertisement, позволяющая анонсировать маршрутную информацию, описанную в advertise-map, в зависимости от условий и наличия в BGP RIB-маршрутов, описанных в condition-map.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
advertise-map <ADVERTISE> {EXIST-MAP | NOT-EXIST-MAP} <CONDITION>
no advertise-map
```

Параметры

<ADVERTISE> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.

<CONDITION> – имя конфигурируемых правил маршрутизации, задаётся строкой до 31 символа.

<EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, присутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.

<NOT-EXIST-MAP> – условие проверки: если маршруты, описанные в condition-map, отсутствуют в BGP RIB, то происходит анонсирование маршрутов, описанных в advertise-map.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-NEIGHBOR-AF

CONFIG-BGP-GROUP-AF

CONFIG-BGP-NEIGHBOR-AF-VRF

CONFIG-BGP-GROUP-AF-VRF

Пример

```
rtt(config-bgp-neighbor-af)# advertise-map ADVERTISE exist-map CONDITION
```

advertise-map

Данная команда позволяет задать дополнительные условия агрегирования маршрутов. Условия задаются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
advertise-map [ route-map <NAME> ]
no advertise-map
```

Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример

```
rtt(config-bgp-aggregate)# advertise-map example-route-map
```

aggregate-address

Данной командой включается функция агрегирования маршрутов, передаваемых по BGP.

Использование отрицательной формы команды (no) отключает данную функцию.

Агрегируемый маршрут будет анонсироваться при соблюдении следующих условий:

- Наличие в таблицы маршрутизации хотя бы одного маршрута, входящего в агрегированный;
- Хотя бы один из маршрутов, входящий в агрегированный, должен быть лучшим в таблице маршрутизации.

Синтаксис

```
[no] aggregate-address { <ADDR/LEN> | <IPV6-ADDR/LEN> }
```

Параметры

<ADDR/LEN> – IP-адрес и маска подсети, задается в виде AAA.BBB.CCC.DDD/EE, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32];

<IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задается в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# aggregate-address 192.168.0.0/16
```

allow-local-as

Данной командой задается режим, в котором разрешен приём маршрутов в BGP-атрибуте, AS Path которых содержатся номера автономной системы процесса.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
allow-local-as <NUMBER>
no allow-local-as
```

Параметры

<NUMBER> – пороговое число вхождений номера автономной системы процесса в атрибуте AS Path, при которых маршрут будет принят, диапазон допустимых значений [1..10].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# allow-local-as
```

as-range

Данная команда позволяет указать номера AS, с которыми будет динамически устанавливаться BGP-сессия.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
as-range <AS-PATH>
```

no as-range <AS-PATH>

Параметры

<AS-PATH> – список номеров автономных систем, задается в виде AS-AS,AS,AS-AS, принимает значения [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-LISTEN

CONFIG-BGP-VRF-LISTEN

Пример

```
rtt(config-bgp-listen)# as-range 65000-65500,65527
rtt(config-bgp-listen)#
```

as-set

Данной командой задаётся режим, в котором в AS-Path агрегированного маршрута устанавливаются номера AS из AS-Path его компонентов.

Использование отрицательной формы команды (no) отключает данный режим.

Синтаксис

```
[no] as-set
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример

```
rtt(config-bgp-aggregate)# as-set
```

attribute-map

Данная команда позволяет устанавливать дополнительные атрибуты агрегированного маршрута. Атрибуты указываются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию

Синтаксис

```
attribute-map [ route-map <NAME> ]
no attribute-map
```

Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример

```
rtt(config-bgp-aggregate)# attribute-map example-route-map
```

clear bgp

Данная команда сбрасывает все или определенный BGP-процесс.

Синтаксис

```
clear bgp [ <AS> ] [ vrf <NAME_VRF> ] [ neighbor { <ADDR> | <IPV6_ADDR> } ]
```

Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295];

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6_ADDR> – IPv6-адрес соседа, задается в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<NAME-VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будут сброшены сессии протокола BGP.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear bgp
rtt# clear bgp 1000
```

cluster-id

Командой устанавливается идентификатор Route-Reflector кластера, которому принадлежит BGP-процесс маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

Синтаксис

```
cluster-id <ID>
```

```
no cluster-id
```

Параметры

<ID> – идентификатор Route-Reflector кластера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp-af)# cluster-id 1.1.1.1
```

confederation identifier

Командой устанавливается идентификатор конфедерации для инстанса BGP.

Использование отрицательной формы команды (no) удаляет идентификатор.

Синтаксис

```
confederation identifier <ID>
no confederation identifier
```

Параметры

<ID> – идентификатор конфедерации, принимает значение [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# confederation identifier 65514
```

confederation peer

Командой задаются участники конфедерации для инстанса BGP.

Использование отрицательной формы команды (no) удаляет участника.

Синтаксис

```
confederation peer <AS>
no confederation peer <AS>
```

Параметры

<AS> – список номеров автономных систем, задается в виде AS-AS,AS,AS-AS, принимает значения [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# confederation peer 65515
```

connection-mode passive

Данной командой устанавливается пассивный режим установки соединения с BGP-пиром. Маршрутизатор не инициирует установку BGP-соседства и принимает только входящие соединения.

Использование отрицательной формы команды (no) устанавливает режим по умолчанию.

Синтаксис

```
[no] connection-mode passive
```

Параметры

Команда не содержит параметров

Значение по умолчанию

Маршрутизатор инициирует установку BGP соседства и принимает входящие соединения.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# connection-mode passive
```

default-originate

Данной командой задается режим, в котором BGP-соседу в обновлении на ряду с другими маршрутами отправляется маршрут по умолчанию.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] default-originate
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Пример

```
rtt(config-bgp-neighbor)# default-originate
```

description

Данной командой определяется описание соседа.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание соседа, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# description "ISP_RTK"
```

ebgp-multihop

Данной командой разрешается подключение к соседям, которые находятся в не напрямую подключенных подсетях.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] ebgp-multihop <NUM>
```

Параметры

<NUM> – максимальное количество хопов при установке EBGP (используется для TTL).

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# ebgp-multihop
```

fall-over

Данная команда позволяет активировать один из методов обнаружения неактивного пиринга по протоколу BGP.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] fall-over { <BFD> | route-map <NAME> }
```

Параметры

<BFD> – активация протокола BFD на конфигурируемом BGP-соседе.

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа. Данная настройка включает механизм Fast Peer Deactivation.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# fall-over bfd
```

flow-spec enable

Данной командой устанавливается режим трансляции flow-spec информации при работе с BGP-соседом или группой BGP-соседей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] flow-spec enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Пример

```
rtt(config-bgp-group)# flow-spec enable
```

graceful-restart

Данной командой включается механизм, позволяющий не удалять маршрутную информацию, принятую от пира, при разрыве соседства по истечении заданного времени.

Синтаксис

```
[no] graceful-restart { timeout <TIME> }
```

Параметры

<TIME> – время в секундах, по истечении которого пиринг считается завершённым.

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# graceful-restart
```

inherit-cost ospf

Включает наследование метрики у маршрутов OSPF в атрибут MED при редистрибуции в протокол BGP.

Синтаксис

```
[no] inherit-cost ospf
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# inherit-cost ospf
```

ipv6 router bgp log-neighbor-changes

Данной командой включается логирование изменений состояния IPv6-BGP-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния IPv6-BGP-соседей.

Синтаксис

```
[no] ipv6 router bgp log-neighbor-changes
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 router bgp log-neighbor-changes
```

ipv6 router bgp maximum-paths

Данной командой включается ECMP и определяется максимальное количество равноценных IPv6-маршрутов до цели.

Использование отрицательной формы команды (no) отключает ECMP.

Синтаксис

```
ipv6 router bgp maximum-paths <VALUE>
no ipv6 router bgp maximum-paths
```

Параметры

<VALUE> – количество допустимых равноценных IPv6-маршрутов до цели, принимает значения [1..16].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 router bgp maximum-paths 14
```

listen-range

Данная команда позволяет динамически устанавливать BGP-сессию с маршрутизаторами, которые имеют IP-адреса из указанной подсети, без указания конкретного адреса соседа.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
listen-range { <ADDR/LEN> | <IPV6-ADDR/LEN }
```

no listen-range { <ADDR/LEN> | <IPV6-ADDR/LEN }

Параметры

<IP-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IP-ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# listen-range 192.168.0.0/16
rtt(config-bgp-listen)#
```

local-as

Данная команда позволяет маршрутизатору представляться номером другой автономной системы в дополнение к своей реальной AS.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
local-as <AS>
```

no local-as

Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# local-as 64500
rtt(config-bgp-local-as)#
```

neighbor

Данной командой добавляется BGP-сосед и осуществляется переход в режим настройки параметров BGP-соседа.

Использование отрицательной формы команды (no) удаляет параметры соседнего маршрутизатора из конфигурации.

Синтаксис

```
[no] neighbor {<ADDR> |<IPV6-ADDR>}
```

Параметры

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# neighbor 192.168.0.2
rtt(config-bgp-neighbor)#
```

next-hop resolve-recursive

Данной командой устанавливается глубина рекурсивного поиска по next-hop атрибуту маршрута, полученного по протоколу BGP. Если поиск завершится безрезультатно, то маршрут будет помечен как "unreachable".

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
next-hop-resolve-recursive {<VALUE>}
no next-hop-resolve-recursive
```

Параметры

<VALUE> – глубина поиска, принимает значения [1..10].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-FAMILY

CONFIG-BGP-VRF-FAMILY

Пример

```
rtt(config-bgp-af)# next-hop resolve-recursive 10
```

next-hop-self

Данной командой задается режим, в котором для всех маршрутов, полученных от eBGP-пиров, атрибут next-hop будет замен на адрес локального маршрутизатора при отправке BGP-пирам.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
next-hop-self [all]

no next-hop-self
```

Параметры

all – для eBGP- и iBGP-пиров в анонсируемой маршрутной информации в атрибуте next-hop будет указан адрес локального маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-GROUP-AF

CONFIG-BGP-GROUP-AF-VRF

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-AF

CONFIG-BGP-NEIGHBOR-AF-VRF

Пример

```
rtt(config-bgp-neighbor-af)# next-hop-self
```

no-prepend

Данная команда исключает добавление номера AS, указанного в опции local-as, в AS Path принимаемых маршрутов.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] no-prepend
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-LOCAL-AS

CONFIG-BGP-VRF-LOCAL-AS

Пример

```
rtt(config-bgp-local-as)# no-prepend
```

peer-group

Данной командой создается группа BGP-соседей с заданным именем и осуществляется переход в режим конфигурирования параметров группы.

Использование отрицательной формы команды (no) удаляет BGP-группу с заданным именем.

Синтаксис

```
[no] peer-group <NAME>
```

Параметры

<NAME> – название группы, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp)# peer-group list1
```

peer-group

Данной командой к BGP-группе или BGP-соседу применяются настройки, описанные в заданной BGP-группе.

Использование отрицательной формы команды (no) отменяет применение настроек, описанных в заданной BGP-группе.

Синтаксис

```
peer-group <NAME>
no peer-group
```

Параметры

<NAME> – название группы, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-af)# peer-group list1
```

preference

Данная команда определяет приоритетность маршрутов, получаемых от соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
preference <VALUE>
no preference
```

Параметры

<VALUE> – приоритетность маршрутов соседа, принимает значения в диапазоне [1..255].

Значение по умолчанию

170

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Пример

```
rtt(config-bgp-neighbor)# preference 30
```

remote-as

Данной командой устанавливается номер автономной системы BGP-соседа.

Использование отрицательной формы команды (no) удаляет номер автономной системы.

Синтаксис

```
remote-as <AS>
no remote-as
```

Параметры

<AS> – номер автономной системы, принимает значения [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# remote-as 20
```

remove-private-as

Данной командой задается режим, в котором перед отправлением обновления из BGP-атрибута AS Path маршрутов удаляются приватные номера автономных систем (в соответствии с RFC 6996).

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] remove-private-as <ACTION>
```

Параметры

<ACTION> – действие, указывающее на то, какие приватные AS должны быть удалены или заменены. Принимает одно из значений:

- all – удалить все включения приватных AS из оригинального AS path;
- nearest – удалить все приватные AS до последней (правой) публичной AS в оригинальном AS path;
- replace – заменить все приватные AS на номер AS, в процессе которого обрабатывается данная команда.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Пример

```
rtt(config-bgp-neighbor-af)# remove-private-as
```

replace-as

Данная команда исключает добавление в AS Path анонсируемого маршрута номера реальной AS.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] replace-as
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-LOCAL-AS

CONFIG-BGP-VRF-LOCAL-AS

Пример

```
rtt(config-bgp-local-as)# no-prepend
```

router bgp

Данной командой добавляется BGP-процесс в систему и осуществляется переход в режим настройки параметров BGP-процесса.

Использование отрицательной формы команды (no) удаляет BGP-процесс из системы.

Синтаксис

```
[no] router bgp <AS> [ unit <ID> ]
```

Параметры

<AS> – номер автономной системы процесса, принимает значения [1..4294967295];

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router bgp 1000
rtt(config-bgp)#
```

Добавлен BGP-процесс с автономной системой 1000.

router bgp log-neighbor-changes

Данной командой включается логирование изменений состояния BGP-соседей.

Использование отрицательной формы команды (no) отключает логирование изменений состояния BGP-соседей.

Синтаксис

```
[no] router bgp log-neighbor-changes
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router bgp log-neighbor-changes
```

router bgp maximum-paths

Данной командой включается ECMP и определяется максимальное количество равноценных маршрутов до цели.

Использование отрицательной формы команды (no) отключает ECMP.

Синтаксис

```
router bgp maximum-paths <VALUE>
no router bgp maximum-paths
```

Параметры

<VALUE> – количество допустимых равноценных маршрутов до цели, принимает значения [1..16].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router bgp maximum-paths 14
```

route-reflector-client

Данной командой указывается, что BGP-сосед является Route-Reflector клиентом.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
[no] route-reflector-client
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# route-reflector-client
```

router-id

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

Синтаксис

```
router-id { <ID> | <IF> | <TUN> }
no router-id
```

Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-BGP-VRF

Пример

```
rtt(config-bgp-af)# router-id 1.1.1.1
```

send-community extended

Включает отправку параметра extended community в отсылаемых соседу анонсах. По умолчанию параметры extended community удаляются из отправляемых анонсов. Для корректной работы VPNv4/L2VPN отправку extended community необходимо включать.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] send-community extended
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Значение по умолчанию

Запрещено.

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR-FAMILY

CONFIG-BGP-VRF-NEIGHBOR-FAMILY

Пример

```
rtt(config-bgp-neighbor-af)# send-community extended
```

show bgp flow-spec

Данная команда отображает информацию о правилах flow-spec.

Синтаксис

```
show bgp [ vrf <VRF-NAME> ] flow-spec [ neighbor <ADDR> ] [ instance <AS-NUM> ]
```

Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<ADDR> – IP-адрес, который задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show bgp flow-spec
--------------------------------------------------------------------------------
```

```
Destination Prefix: 192.168.54.2/32
IP Protocol:        17
Ext-community type: traffic-rate (0x8006)
```

show bgp ipv4 unicast

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

Синтаксис

```
show bgp [ vrf <VRF-NAME> ] ipv4 unicast [ { <IP-ADDR> | <IP-ADDR/LEN> } ] [
instance <AS-NUM> ] [ neighbor <IP-ADDR> [ { advertise-routes | routes } ] ]
```

Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295];

<IP-ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IP-ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

routes – при указании команды отображается маршрутная информация, полученная от соседа;

advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу. Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show bgp ipv4 unicast
Status codes: u - unicast, b - broadcast, m - multicast a - anycast
              * - valid, > - best
Origin codes: i - IGP, e - EGP, ? - incomplete
     Network              Next Hop             Metric  LocPrf  Path
*> u 10.0.10.0/24         10.115.0.1                   100      i
*> u 0.0.0.0/0            10.115.0.1                   100      i
*  u 14.0.10.0/24         10.115.0.1                   100      i
```

Пример 2

```
rtt# show bgp ipv4 unicast 75.0.0.0
75.0.0.0/24        via 115.0.0.40 on gi1/0/14        [bgp20 2000-01-15] (AS90?)
    Administrative Distance: 68
    Type:                    unicast
    Origin:                  Incomplete
    AS PATH:                 1 30 70 90
    Next Hop:                115.0.0.40
    MED:                     0
    Local Preference:        100
    Community:               (1:555)
    Valid, Best
```

show bgp ipv6 unicast

Данная команда отображает таблицу маршрутизации BGP или детальную информацию об определенном маршруте при использовании фильтров.

Синтаксис

```
show bgp [ vrf <VRF-NAME> ] ipv6 unicast [ { <IPV6-ADDR> | <IPV6-ADDR/LEN> } ] [
instance <AS-NUM> ] [ neighbor <IPV6-ADDR> [ { advertise-routes | routes } ] ]
```

Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IPV6-ADDR/LEN> – подсеть, задаётся в виде X:X:X:X::X/EE, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF],EE принимает значения [1..128];

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации;

routes – при указании команды отображается маршрутная информация, полученная от соседа;

advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt#  show bgp ipv6 unicast
```

```
  Status codes: u - unicast, b - broadcast, m - multicast, a - anycast
                * - valid, > - best
  Origin codes: i - IGP, e - EGP, ? - incomplete

       Network              Next Hop             Metric  LocPrf     Weight Path
  *> u 2600:380:180::/41    fc00:0:1409:2800::1  --      100        0      65054 12389 3356 ... i
  *> u 2804:2404:8000::/34  fc00:0:1409:2800::1  --      100        0      65054 12389 266925 ... i
  *> u 2400:3800:8800::/37  fc00:0:1409:2800::1  --      100        0      65054 12389 2497 ... i
  *> u 2400:54c0:c0::/44    fc00:0:1409:2800::1  --      100        0      65054 12389 3356 ... i
```

Пример 2

```
rtt# show bgp ipv6 unicast 2600:380:180::/41
 2600:380:180::/41  via fc00:0:1409:2800::1 on gi1/0/1.2800 [bgp65514 14:08:30] (20057i)
     Administrative Distance: 170
     Type:                    unicast
     Origin:                  IGP
     AS PATH:                 65054 12389 3356 7018 20057
     Next Hop:                fc00:0:1409:2800::1 (fe80::aaf9:4bff:feaa:2dd5)
     Local Preference:        100
     Community:               (3356:3) (3356:22) (3356:86) (3356:575) (3356:666) (3356:903) (3356:2059)
                              (12389:6)
     Weight:                  0
     Valid, Best
```

show bgp l2vpn vpls

Данная команда отображает информацию о всех принятых l2vpn-маршрутах.

Синтаксис

```
show bgp l2vpn vpls { all | rd <ASN:nn> } [ neighbor <ADDR> { advertise-routes |
routes ] [ ve-id <ID> block-offset <ID> ]
```

Параметры

- all – при указании команды all отображается полная информация l2vpn-маршрутов;
- rd – при указании команды rd отображается информация о маршрутах с заданным RD;
- routes – при указании команды отображается маршрутная информация, полученная от соседа;
- advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу;
- <RD> – значение Route distinguisher, задается в одном из следующем виде:
    - <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
    - <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
    - <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].
- <IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
- <ID> – принимает значение [1..65535], задает идентификатор vpls edge или block-offset.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT# show bgp l2vpn vpls all
 Status codes: * - valid, > - best, i - internal, S - stale
 Origin codes: i - IGP, e - EGP, ? - incomplete

 Codes Route Distinguisher   VID   VBO   VBS   Next hop        Metric     LocPrf     Weight Path
 ----- --------------------- ----- ----- ----- --------------- ---------- ---------- ------ -------------------
 *>i   65502:100             3     1     8     192.168.9.9         --         100        0      i
 *>    65502:102             2     1     10    --                  --         --         --
 *>    65502:100             2     1     10    --                  --         --         --
 *>i   65502:102             1     1     10    192.168.7.7         --         100        0      i
 *>i   65502:100             1     1     10    192.168.7.7         --         100        0      i
```

show bgp neighbors

Данная команда отображает информацию о всех или о выбранном BGP-пире.

Синтаксис

```
show bgp [vrf <VRF-NAME>] neighbors [instance <AS-NUM> ] { <ADDR> | <IPV6-ADDR> }
```

Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

<ADDR> – IP- или IPv6-адрес пира, который задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], или в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<IPV6-ADDR/LEN> – подсеть, задаётся в виде X:X:X:X::X/EE, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF],EE принимает значения [1..128].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show bgp neighbors
BGP neighbor is 10.115.0.1
    BGP state:        Established
    Neighbor address: 10.115.0.1
    Neighbor AS:      20
    Neighbor ID:      115.0.0.1
    Neighbor caps:    refresh restart-aware AS4
    Session:          internal multihop AS4
    Source address:   10.115.0.2
    Hold timer:       137/180
    Keepalive timer:  10/60
   Incoming prefix-list: from_ISP
    Outgoing prefix-list: to_ISP
    Incoming route-map:   comingS
    Outgoing route-map:   AS_prepend
    Uptime:               12 s
    BFD address:          192.168.1.2
    BFD state:            Up
    BFD interval:         3.000 s
    BFD timeout:          15.000 s
```

Пример 2

```
rtt#  show bgp neighbors 100.64.28.1
BGP neighbor is 100.64.28.1
    BGP state:                        Established
    Neighbor address:                 100.64.28.1
    Neighbor AS:                      65054
    Neighbor ID:                      5.5.0.111
    Neighbor caps:                    refresh enhanced-refresh restart-aware AS4
    Session:                          external AS4
    Source address:                   100.64.28.160
    Weight:                           0
    Hold timer:                       165/180
    Keepalive timer:                  38/60
    RR client:                        No
    Address family ipv4 unicast:
      Default originate:              No
      Default information originate: No
      Outgoing route-map:             RouteMap_out
      Preference:                     170
      Remove private AS:              No
      Next-hop self:                  No
    Uptime:                           12952 s
```

Пример 3

```
rtt# show bgp neighbors fc00:0:1409:2800::1
BGP neighbor is fc00:0:1409:2800::1
    BGP state:                        Established
    Neighbor address:                 fc00:0:1409:2800::1
    Neighbor AS:                      65054
    Neighbor ID:                      5.5.0.111
    Neighbor caps:                    refresh enhanced-refresh restart-aware AS4
    Session:                          external AS4
    Source address:                   fc00:0:1409:2800::16
    Weight:                           0
    Hold timer:                       174/180
    Keepalive timer:                  50/60
    RR client:                        No
    Address family ipv6 unicast:
      Default originate:              No
      Default information originate: No
```

```
      Incoming route-map:           RouteMap_in
      Outgoing route-map:           RouteMap_out
      Preference:                   170
      Remove private AS:            No
      Next-hop self:                No
    Uptime:                         13003 s
```

show bgp summary

Данная команда отображает информацию о состоянии соединений с BGP-пирами.

Синтаксис

```
show bgp [vrf <VRF-NAME>] summary [instance <AS-NUM>]
```

Параметры

<AS-NUM> – номер автономной системы процесса, принимает значения [1..4294967295].

<VRF-NAME> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show bgp summary
 Tue Sep 14 15:36:26 2021
   BGP router identifier 100.64.28.160, local AS number 65514
   BGP activity 984813/0 prefixes
   Neighbor                AS             MsgRcvd      MsgSent      Up/Down        St/PfxRcd
                                                                    (d,h:m:s)
   ----------------------  -------------  ----------   ----------   ----------     ------------
   10.20.0.2               65058          0            0            00,01:28:02     Idle
   100.64.28.1             65054          861119       102          00,01:27:56     855897
```

show bgp vpnv4 unicast

Данная команда отображает информацию о vpnv4-маршрутах. Указание префикса в команде отображает детальную информацию по префиксу.

Синтаксис

```
show bgp vpnv4 unicast { all | rd <RD> | vrf <NAME_VRF> } [ neighbor <IP> { routes
| advertise-routes } ] [ <PREFIX> ]
```

Параметры

all – при указании команды all отображается полная информация vpnv4-маршрутов;
rd – при указании команды rd отображается информация о маршрутах с заданным RD;
vrf – при указании команды vrf отображается информация vpnv4-маршрутов для указанного VRF;
routes – при указании команды отображается маршрутная информация, полученная от соседа;
advertise-routes – при указании команды отображается маршрутная информация, объявленная соседу;
<RD> – значение Route distinguisher, задается в одном из следующем виде:

- <ASN>:<nn> – где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn> – где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4BASN>:<nn> – где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535];
- <NAME_VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
<PREFIX> – префикс, имеет вид – AAA.BBB.CCC.DDD/EE, где AAA-DDD принимают значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show bgp vpnv4 unicast all
 Status codes: * - valid, > - best, i - internal, S - stale
 Origin codes: i - IGP, e - EGP, ? - incomplete

 Codes   Route Distinguisher   IP Prefix            Next hop         Metric   Label     LocPrf   Weight Path
 -----  ---------------------  ------------------ --------------- ---------- ------- ---------- ------ ----------------
 *>i         65000:10001      10.100.106.1/32        6.6.6.6           --       16        100      0      ?
 *>          65000:10002      10.20.0.0/24           --                --       17         --       --     ?
 *>          65000:10002      10.20.1.0/24           --                --       17         --       --     ?
 *>i         65000:10001      10.100.0.4/30           2.2.2.2          --       16        100       0      i
 *>          65000:10002      172.18.0.0/30           --               --       17        --        --     ?
 *>i         65000:10002      10.30.0.0/24            5.5.5.5          --       17        100       55     i
 *>i         65000:10001      10.100.104.0/24     2.2.2.2           --       16       100       0      i
 *>          65000:10001      10.100.105.0/24         --               --       16        --        --     ?
 *>i         65000:10001      10.100.106.0/24     6.6.6.6           --       16       100       0      ?
 *>          65000:10001      203.203.203.203/32      --               --       16        --        --     ?
 *>          65000:10001      202.202.202.202/32      --               --       16        --        --     ?
 *>i         65000:10001      10.100.100.0/24        2.2.2.2        --       16       100       0      i
 *>          65000:10001      10.100.0.0/30          --               --       16        --        --     ?
 *>          65000:10001      10.100.102.0/24        --               --       16        --        --     ?
```

Пример 2

```
rtt# show bgp vpnv4 unicast all 10.100.106.0/24
```

```
BGP routing table entry for 65000:10001 10.100.106.0/24
Next hop: 6.6.6.6
Label: 16
AS path: --
Origin: Incomplete
Local preference: 100
Extended Community: RT:65000:10001
Weight: 0
```

summary-only

Данной командой задаётся режим, в котором будут подавляться все компоненты агрегированного маршрута.

Использование отрицательной формы команды (no) отключает данный режим.

Синтаксис

```
[no] summary-only
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример

```
rtt(config-bgp-aggregate)# summary-only
```

suppress-map

Данная команда позволяет подавлять компоненты агрегированного маршрута. Параметры для подавления указываются через route-map.

Использование отрицательной формы команды (no) отключает данную функцию.

Синтаксис

```
suppress-map [ route-map <NAME> ]
no suppress-map
```

Параметры

<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-AGGREGATE

CONFIG-BGP-VRF-AGGREGATE

Пример

```
rtt(config-bgp-aggregate)# suppress-map example-route-map
```

timers error-wait

Данной командой устанавливается время минимальной и максимальной задержки, в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers error-wait <TIME1> <TIME2>
no timers error-wait
```

Параметры

<TIME1> – время в секундах, принимает значения [1..65535];

<TIME2> – время в секундах, принимает значения [1..65535].

Значение по умолчанию

60 и 300

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp)# timers error-wait 90 450
```

timers holdtime

Данной командой устанавливается временной интервал, по истечении которого встречная сторона считается недоступной. Таймер запускается после установления отношений соседства и начинает отсчёт от 0. Таймер сбрасывается при получении каждого ответа на keepalive-сообщение от встречной стороны. Рекомендуется устанавливать значение таймера равное *3 * keepalive*.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers holdtime <TIME>
no timers holdtime
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

180

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-af)# timers holdtime 360
```

timers keepalive

Данной командой устанавливается временной интервал, по истечении которого идет проверка соединения со встречной стороной.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers keepalive <TIME>
no timers keepalive
```

Параметры

<TIME> – время в секундах, принимает значения [1..32767].

Значение по умолчанию

60

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

CONFIG-VRF-BGP

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-af)# timers keepalive 120
```

update-source

Данной командой определяется IP/IPv6-адрес маршрутизатора, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых обновлениях маршрутной информации BGP.

Использование отрицательной формы команды (no) удаляет указанный IP/IPv6-адрес источника.

Синтаксис

```
update-source { <ADDR> | <IPV6-ADDR> | <IF> | <TUN> } [ unit <ID> ]
no source-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<ID> – номер юнита, принимает значения [1..4].

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# update-source 10.100.100.2
```

vrf

Данной командой в системе создается экземпляр BGP процесса в VRF и осуществляется переход в режим настройки параметров BGP процесса в указанном VRF.

Использование отрицательной формы команды (no) удаляет экземпляр BGP VRF из системы.

Синтаксис

```
[no] ip vrf <VRF>
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP

Пример

```
rtt(config-bgp)# vrf xx
rtt(config-bgp-vrf)#
```

weight

Данной командой устанавливается вес маршрутов, принимаемый от данного BGP-соседа или группы BGP-соседей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
weight <WEIGHT>
no weight
```

Параметры

<WEIGHT> – значение веса маршрута, принимает значения [0..65535].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-BGP-GROUP

CONFIG-BGP-VRF-GROUP

CONFIG-BGP-NEIGHBOR

CONFIG-BGP-VRF-NEIGHBOR

Пример

```
rtt(config-bgp-neighbor)# weight 200
```

21.8. Настройка протоколов RIP и RIPNG

authentication key-chain

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

Синтаксис

```
authentication key-chain <KEYCHAIN>
no authentication key-chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

Пример

```
rtt(config-rip)# authentication key-chain lock
```

clear ip rip

Данная команда удаляет содержимое базы маршрутов RIP.

Синтаксис

```
clear ip rip
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip rip
```

clear ipv6 rip

Данная команда удаляет содержимое базы маршрутов RIPNG.

Синтаксис

```
clear ipv6 rip
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ipv6 rip
```

ip rip bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола RIP.

Использование отрицательной формы команды (no) отключает на интерфейсе протокол BFD для протокола RIP.

Синтаксис

```
[no] ip rip bfd-enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Протокол BFD для протокола RIP отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip rip bfd-enable
```

ip rip metric

Данная команда устанавливает величину метрики на интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

Синтаксис

```
ip rip metric <VALUE>
no ip rip metric
```

Параметры

<VALUE> – величина метрики, задаётся в размере [1..15].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip rip metric 11
```

ip rip mode

Данная команда устанавливает режим анонсирования маршрутов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip rip mode <MODE>
```

```
no ip rip mode
```

Параметры

<MODE> – режим анонсирования маршрутов:

- multicast – маршруты анонсируются в многоадресном режиме;
- broadcast – маршруты анонсируются в широковещательном режиме;
- unicast – маршруты анонсируются в unicast-режиме соседям, настроенным с помощью команды *ip rip neighbor <ADDR>*.

Значение по умолчанию

multicast

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip rip mode broadcast
```

ip rip neighbor

Данной командой статически задается IP-адрес соседа для установления отношения в unicast-режиме анонсирования маршрутов.

Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

Синтаксис

```
[no] ip rip neighbor <ADDR>
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip rip neighbor 10.100.100.5
```

ip rip summary-address

Данной командой включается суммаризация подсетей.

Использование отрицательной формы команды (no) отключает суммаризацию подсетей.

Синтаксис

```
[no] ip rip summary-address <ADDR/LEN>
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip rip summary-address 10.200.200.0/24
```

ipv6 rip metric

Данная команда устанавливает величину метрики на интерфейсе.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

Синтаксис

```
ipv6 rip metric <VALUE>
no ipv6 rip metric
```

Параметры

<VALUE> – величина метрики, задаётся в размере [1..15].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 rip metric 11
```

ipv6 rip summary-address

Данной командой включается суммаризация подсетей IPv6.

Использование отрицательной формы команды (no) отключает суммаризацию подсетей.

Синтаксис

```
[no] ipv6 rip summary-address <ADDR/LEN>
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 rip summary-address fb00::1/20
```

ipv6 router rip

Данной командой осуществляется переход в режим настройки параметров RIPNG-процесса.

Использование отрицательной формы команды (no) устанавливает параметры RIPNG-процесса на значения по умолчанию.

Синтаксис

```
[no] ipv6 router rip
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 router rip
rtt(config-ripng)#
```

passive-interface

Данная команда выключает анонсирования маршрутов по интерфейсу.

Использование отрицательной формы команды (no) восстанавливает анонсирования маршрутов.

Синтаксис

```
[no] passive-interface { <IF> | <TUN> }
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

Пример

```
rtt(config-rip)# passive-interface gigabitethernet 1/0/15
```

passive-interface

Данная команда выключает анонсирования маршрутов по интерфейсу.

Использование отрицательной формы команды (no) восстанавливает анонсирования маршрутов.

Синтаксис

```
[no] passive-interface <IF>
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIPNG

Пример

```
rtt(config-ripng)# passive-interface gigabitethernet 1/0/15
```

poison-reverse

Данная команда позволяет получать маршруты от соседа и отправлять обратно с метрикой 16.

Синтаксис

```
ip rip poison-reverse
no ip rip poison-reverse
```

```
ipv6 rip poison-reverse
no ipv6 rip poison-reverse
```

Значение по умолчанию

ip rip poison-reverse

ipv6 rip poison-reverse

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

```
rtt(config-if-gi)# no ip rip poison-reverse
rtt(config-if-gi)# no ipv6 rip poison-reverse
```

router rip

Данной командой осуществляется переход в режим настройки параметров RIP-процесса.

Использование отрицательной формы команды (no) устанавливает параметры RIP-процесса на значения по умолчанию.

Синтаксис

```
[no] router rip
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router rip
rtt(config-rip)#
```

show ip rip

Данная команда отображает таблицу маршрутизации RIP. Отображаются все маршруты, полученные по протоколу RIP

Синтаксис

```
show ip rip
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

show ipv6 rip

Данная команда отображает таблицу маршрутизации RIPNG. Отображаются все маршруты, полученные по протоколу RIP.

Синтаксис

```
show ipv6 rip
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 rip
10.10.0.1/32    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
10.1.90.0/24    via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
192.168.16.0/24 via 115.0.0.10 on gi1/0/15 [rip 21:31:17] * (100/6)
```

split-horizon

Данная команда позволяет получать маршруты от соседа и не отправлять обратно.

Синтаксис

```
ip rip split-horizon
no ip rip split-horizon

ipv6 rip split-horizon
no ipv6 rip split-horizon
```

Значение по умолчанию

ip rip split-horizon

ipv6 rip split-horizon

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# no ip rip split-horizon
rtt(config-if-gi)# no ipv6 rip split-horizon
```

timers flush

Данной командой устанавливается временной интервал, по истечении которого производиться удаление маршрута. При установке значения нужно учитывать следующее правило: «timers invalid + 60».

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers flush <TIME>
no timers flush
```

Параметры

<TIME> – время в секундах, принимает значения [12..65535].

Значение по умолчанию

240 секунд.

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-rip)# timers flush 300
```

timers invalid

Данной командой определяется временной интервал корректности маршрутной записи без обновления. По истечению срока, без получения обновления, маршрут помечается как недоступный.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers invalid <TIME>
no timers invalid
```

Параметры

<TIME> – время в секундах, принимает значения [12..65535].

Значение по умолчанию

180

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-rip)# timers invalid 240
```

timers update

Данной командой устанавливается временной интервал, по истечении которого производится анонсирование.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timers update <TIME>
```

```
no timers update
```

Параметры

<TIME> – время в секундах, принимает значения [12..65535].

Значение по умолчанию

30

Необходимый уровень привилегий

10

Командный режим

CONFIG-RIP

CONFIG-RIPNG

Пример

```
rtt(config-rip)# timers update 25
```

21.9. Настройка протоколов OSPF и OSPFv3

area

Данной командой устанавливается идентификатор области.

Использование отрицательной формы команды (no) удаляет созданную область.

Синтаксис

```
[no] area <AREA_ID>
```

Параметры

<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример 1

```
rtt(config-ospf)# area 11.11.11.51
```

Пример 2

```
rtt(config-ipv6-ospf)# area 11.11.11.51
```

area-type

Команда определяет тип области.

Использование отрицательной формы команды (no) устанавливает тип области как стандартный.

Синтаксис

```
[no] area-type <TYPE> [ no-summary ]
```

Параметры

<TYPE> – тип области:

- normal – устанавливает значение standart (стандартная область);
- stub – устанавливает значение stub (тупиковая область).

no-summary – команда в связке с параметром «stub» образует область «totally stubby» (для передачи информации за пределы области используется только маршрут по умолчанию).

- nssa – устанавливает значение nssa (область NSSA).

no-summary – в связке с параметром nssa образует область totally nssa (автоматически генерирует маршрут по умолчанию как межобластной).

Значение по умолчанию

normal

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример 1

```
rtt(config-ospf-area)# area-type stub
```

Пример 2

```
rtt(config-ipv6-ospf-area)# area-type nssa
```

authentication key

Данная команда определяет пароль для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером до 8 байт (от 2 до 16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-OSPF-VLINK

Пример

```
rtt(config-ospf-vlink)# authentication key ascii-text 12345678
```

authentication key chain

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

Синтаксис

```
authentication key chain <KEYCHAIN>
no authentication key chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-OSPF-VLINK

Пример

```
rtt(config-ospf-vlink)# authentication key chain key2
```

auto-cost reference-bandwidth

Команда задает референсное значение для расчета стоимости интерфейса (cost). Стоимость рассчитывается как отношение reference-bandwidth к bandwidth интерфейса. В случае если отношение меньше 1, стоимость принимается равной 1.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
auto-cost reference-bandwidth <VALUE>
no auto-cost reference-bandwidth
```

Параметры

<VALUE> – величина референсного значения, задаётся в размере [1K..100000000K]. Для удобства конфигурирования, допускается после численного значения вместо литеры 'K' (Kbps) использовать литеру 'M' (Mbps) или 'G' (Gbps).

Значение по умолчанию

100000K

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример

```
rtt(config-ospf)# auto-cost reference-bandwidth 10000K
```

default-cost

Данная команда устанавливает стоимость маршрута по умолчанию, анонсированного в соответствующую зону (при включении "default-information-originate").

Использование отрицательной формы команды (no) возвращает стоимость маршрута в значение по умолчанию.

Синтаксис

```
default-cost <VALUE>
no default-cost
```

Параметры

<VALUE> – стоимость маршрута, принимает значения [1..65535].

Значение по умолчанию

1000

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример

```
rtt(config-ospf-area)# default-cost 100
```

default-information-originate

Данная команда позволяет включить анонсирование маршрута по умолчанию для соответствующей зоны. Применимо для следующих типов зон: Stub, Totally Stub, NSSA, Totally NSSA.

Для Stub, Totally Stub, Totally NSSA зон будет анонсирован маршрут типа IA (inter area).

Для NSSA возможно анонсирование маршрута типа external type-2 или type-1 (по умолчанию).

Использование отрицательной формы команды (no) выключает анонсирование маршрута по умолчанию.

Синтаксис

```
[no] default-information-originate
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример

```
rtt(config-ospf-area)# default-information-originate
```

default-metric-type

Данная команда устанавливает тип маршрута по умолчанию, анонсированного в соответствующей NSSA-зоне (при включении "default-information-originate").

Использование отрицательной формы команды (no) устанавливает тип маршрута в значение по умолчанию.

Синтаксис

```
default-metric-type { type-1 | type-2 }
no default-metric-type
```

Параметры

- type-1 – external-маршрут type-1 (E1);
- type-2 – external-маршрут type-2 (E2).

Значение по умолчанию

type-1

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример

```
rtt(config-ospf-area)# default-metric-type type-2
```

clear ip ospf

Данная команда сбрасывает все или определенный OSPF-процесс.

Синтаксис

```
clear ip ospf [ <ID> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPF-процесс, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip ospf
rtt# clear ip ospf 1000
```

clear ipv6 ospf

Данная команда сбрасывает все или определенный OSPFv3-процесс.

Синтаксис

```
clear ipv6 ospf [ <ID> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будут сброшены все или определенный OSPFv3-процесс, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ipv6 ospf
rtt# clear ipv6 ospf 1000
```

compatible rfc1583

Включается совместимость с RFC 1583.

Использование отрицательной формы команды (no) отключает совместимость с RFC 1583.

Синтаксис

```
[no] compatible rfc1583
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример

```
rtt(config-ospf)# compatible rfc1583
```

dead-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению «hello-interval». Как правило, «dead-interval» равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
dead-interval <TIME>
no dead-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

40

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Пример

```
rtt(config-ospf-vlink)# dead-interval 60
```

hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
hello-interval <TIME>
no hello-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Пример

```
rtt(config-ospf-vlink)# hello-interval 8
```

inherit-metric bgp

Включает наследование внутренней метрики протоколов маршрутизации в метрику BGP, при редистрибуции маршрутов в OSPF.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
[no] inherit-metric bgp
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример 1

```
rtt(config-ospf)# inherit-metric bgp
```

ip ospf

Данной командой включают маршрутизацию по протоколу OSPF на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPF на интерфейсе.

Синтаксис

```
[no] ip ospf
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf
```

ip ospf area

Данная команда определяет принадлежность интерфейса к определенной области OSPF-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPF-процесса.

Синтаксис

```
ip ospf area <AREA_ID>
no ip ospf area
```

Параметры

<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

0.0.0.0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-ip4ip4)# ip ospf area 1.1.1.1
```

ip ospf authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
ip ospf authentication algorithm <ALGORITHM>
no ip ospf authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf authentication algorithm cleartext
```

ip ospf authentication key

Данная команда устанавливает пароль для аутентификации с соседом при передаче пароля открытым текстом.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
ip ospf authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ip ospf authentication key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером до 8 байт (от 2 до 16 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf authentication key ascii-text 123456789
rtt(config-if-gi)# ip ospf authentication key ascii-text encrypted
CDE65039E5591FA3F1
```

ip ospf authentication key-chain

Данная команда определяет список паролей для аутентификации через алгоритм хеширования md5 с соседом.

Использование отрицательной формы команды (no) удаляет привязку к списку паролей.

Синтаксис

```
ip ospf authentication key-chain <KEYCHAIN>
no ip ospf authentication key-chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf authentication key-chain lock
```

ip ospf bandwidth

Данная команда устанавливает величину пропускной способности интерфейса для автоматического расчета стоимости (cost).

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

Синтаксис

```
ip ospf bandwidth <VALUE>
no ip ospf bandwidth
```

Параметры

<VALUE> – величина пропускной способности, задаётся в размере [1K..100000000K]. Для удобства конфигурирования допускается после численного значения вместо литеры 'K' (Kbps) использовать литеру 'M' (Mbps) или 'G' (Gbps).

Значение по умолчанию

Тип интерфейса	Значение по умолчанию
Физический интерфейс	В соответствии с номинальной скоростью (1G для gigabitethernet 10G для tengigabitethernet и т. д.)
SUB и QinQ	Наследуется от родительского физического интерфейса
Bridge	1G
Туннель	100K
Loopback	80G
E1	1544K
Multilink	Определяется как сумма входящих в него интерфейсов умноженная на E1 bandwidth
Port-channel	Определяется как сумма входящих в него интерфейсов умноженная на номинальную скорость Port-channel

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf bandwidth 100M
```

ip ospf bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола OSPF.

Использование отрицательной формы команды (no) отключает на интерфейсе протокол BFD для протокола OSPF.

Синтаксис

```
[no] ip ospf bfd-enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Протокол BFD для протокола OSPF отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf bfd-enable
```

ip ospf cost

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики, рассчитанное автоматически механизмом auto-cost значений ospf bandwidth и auto-cost reference-bandwidth.

Синтаксис

```
ip ospf cost <VALUE> [ track <TRACK-ID> [ default <VALUE> ] ]
no ip ospf cost
```

Параметры

<VALUE> – величина метрики, задаётся в размере [0..32767];

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Значение по умолчанию

Рассчитывается автоматически исходя из значений ospf bandwidth и auto-cost reference-bandwidth.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf cost 11
```

ip ospf dead-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ip ospf dead-interval <TIME>
no ip ospf dead-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

40

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf dead-interval 60
```

ip ospf hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ip ospf hello-interval <TIME>
no ip ospf hello-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1..255].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf hello-interval 8
```

ip ospf instance

Данная команда определяет принадлежность интерфейса к определенному OSPF-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPF-процессу.

Синтаксис

```
ip ospf instance <ID>
no ip ospf instance
```

Параметры

<ID> – номер процесса, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-ip4ip4)# ip ospf instance 300
```

ip ospf mtu-ignore

Данной командой включается режим, в котором OSPF-процесс будет игнорировать значение MTU интерфейса во входящих Database Description-пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

Синтаксис

```
[no] ip ospf mtu-ignore
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf mtu-ignore
```

ip ospf neighbor

Данной командой статически задается IP-адрес соседа для установления отношения в NMBA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

Синтаксис

```
[no] ip ospf neighbor <IP> [ non-eligible ]
```

Параметры

<IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

non-eligible – опциональный параметр, запрещает устройству участвовать в процессе выбора DR в NBMA-сетях. По умолчанию при создании ipv6 ospf neighbor соединение устанавливается в режиме eligible.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf neighbor 10.0.0.2
```

ip ospf network

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip ospf network <TYPE>
no ip ospf network
```

Параметры

<TYPE> – тип сети:

- broadcast – тип соединения широковещательный;
- non-broadcast – тип соединения NBMA;
- point-to-multipoint – тип соединения точка-многоточие;
- point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;

- point-to-point – тип соединения точка-точка.

Значение по умолчанию

broadcast

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf network point-to-point
```

ip ospf passive-interface

Данная команда переводит интерфейс в пассивный режим: с данного интерфейса не будут отправляться hello-пакеты, а также не будут устанавливаться отношения смежности (соседства), однако подсеть, подключенная к такому интерфейсу, будет анонсироваться другим участникам процесса. Использование отрицательной формы команды (no) возвращает интерфейс в режим работы с рассылкой hello-пакетов и установкой соседства.

Синтаксис

```
[no] ip ospf passive-interface
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-TE

CONFIG-TWE

CONFIG-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf passive-interface
```

ip ospf poll-interval

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ip ospf poll-interval <TIME>
no ip ospf poll-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1 .. 255].

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf poll-interval 60
```

ip ospf priority

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

В качестве DR выбирается маршрутизатор с наибольшим приоритетом.

При равных приоритетах в качестве DR выбирается маршрутизатор с наибольшим router id.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip ospf priority <VALUE>
no ip ospf priority
```

Параметры

<VALUE> – приоритет интерфейса, принимает значения [0..255].

Значение по умолчанию

128

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf priority 200
```

ip ospf retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ip ospf retransmit-interval <TIME>
no ip ospf retransmit-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)#ip ospf restransmit-interval 4
```

ip ospf ttl-security-hops

Данной командой определяется значение параметра ttl в исходящих ospf-пакетах. Также во всех входящих ospf-пакетах будет осуществляться проверка, что значение ttl больше либо равно '255 - ttl-security-hops'.

Использование отрицательной формы команды (no) устанавливает значение ttl-security-hops по умолчанию.

Синтаксис

```
ip ospf ttl-security-hops <VALUE>
no ip ospf ttl-security-hops
```

Параметры

<VALUE> – значение ttl [1..255].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-TE

CONFIG-TWE

CONFIG-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

CONFIG-IF-MULTILINK

Пример

```
rtt(config-if-gi)#ip ospf ttl-security-hops 200
```

ip ospf wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ip ospf wait-interval <TIME>
no ip ospf wait-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

40

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ip ospf  wait-interval 60
```

ipv6 ospf

Данной командой включают маршрутизацию по протоколу OSPFv3 на интерфейсе.

Использование отрицательной формы команды (no) отключает маршрутизацию по протоколу OSPFv3 на интерфейсе.

Синтаксис

```
[no] ipv6 ospf
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf
```

ipv6 ospf area

Данная команда определяет принадлежность интерфейса к определенной области OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к определенной области OSPFv3-процесса.

Синтаксис

```
ipv6 ospf area <AREA_ID>
no ipv6 ospf area
```

Параметры

<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-ip4ip4)# ipv6 ospf area 1.1.1.1
```

ipv6 ospf bandwidth

Данная команда устанавливает величину пропускной способности интерфейса для автоматического расчета стоимости (cost).

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

Синтаксис

```
ipv6 ospf bandwidth <VALUE>
ipv6 ip ospf bandwidth
```

Параметры

<VALUE> – величина пропускной способности, задаётся в размере [1K..100000000K]. Для удобства конфигурирования допускается после численного значения вместо литеры 'K' (Kbps) использовать литеру 'M' (Mbps) или 'G' (Gbps).

Значение по умолчанию

Тип интерфейса	Значение по умолчанию
Физический интерфейс	В соответствии с номинальной скоростью (1G для gigabitethernet, 10G для tengigabitethernet и т. д.)
SUB и QinQ	Наследуется от родительского физического интерфейса
Bridge	1G
Туннель	100K
Loopback	80G
E1	1544K
Multilink	Определяется как сумма входящих в него интерфейсов, умноженная на E1 bandwidth
Port-channel	Определяется как сумма входящих в него интерфейсов, умноженная на номинальную скорость Port-channel

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf bandwidth 100M
```

ipv6 osfp bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола OSPFv3.

Использование отрицательной формы команды (no), на интерфейсе отключается протокол BFD для протокола OSPFv3.

Синтаксис

```
[no] ipv6 ospf bfd-enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Протокол BFD для протокола OSPF отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf bfd-enable
```

ipv6 ospf cost

Данная команда устанавливает величину метрики на интерфейсе или туннеле.

Использование отрицательной формы команды (no) устанавливает значение метрики по умолчанию.

Синтаксис

```
ipv6 ospf cost <VALUE> [ track <TRACK-ID> [ default <VALUE> ] ]
no ipv6 ospf cost
```

Параметры

<VALUE> – величина метрики, задаётся в размере [1..65535];

<TRACK-ID> – идентификатор объекта отслеживания, при выполнении всех условий которого будет исполняться указанное действие. Изменяется в диапазоне [1..100].

Значение по умолчанию

Рассчитывается автоматически исходя из значений ospf bandwidth и auto-cost reference-bandwidth.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf cost 11
```

ipv6 ospf dead-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого сосед будет считаться недоступным. Этот интервал должен быть кратным значению hello-interval. Как правило, dead-interval равен 4 интервалам отправки hello-пакетов, то есть 40 секундам.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ipv6 ospf dead-interval <TIME>
no ipv6 ospf dead-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

40

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf dead-interval 60
```

ipv6 ospf hello-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор отправляет следующий hello-пакет.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ipv6 ospf hello-interval <TIME>
no ipv6 ospf hello-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1..255].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf hello-interval 8
```

ipv6 ospf instance

Данная команда определяет принадлежность интерфейса к определенному OSPFv3-процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к OSPFv3-процесс.

Синтаксис

```
ipv6 ospf instance <ID>
no ipv6 ospf instance
```

Параметры

<ID> – номер процесса, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-ip4ip4)# ipv6 ospf instance 300
```

ipv6 ospf mtu-ignore

Данной командой включается режим, в котором OSPFv3-процесс будет игнорировать значение MTU интерфейса во входящих Database Description пакетах.

Использование отрицательной формы команды (no) отключает режим игнорирования MTU интерфейса.

Синтаксис

```
[no] ipv6 ospf mtu-ignore
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf mtu-ignore
```

ipv6 ospf neighbor

Данной командой статически задается IPv6-адрес соседа для установления отношения в NMBA и P2MP (Point-to-MultiPoint) сетях. Использование отрицательной формы команды (no) удаляет статически заданный адрес соседа.

Синтаксис

```
[no] ipv6 ospf neighbor <IPV6-ADDR> [ non-eligible ]
```

Параметры

<IPV6-ADDR> – IPv6-адрес соседа, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

non-eligible – опциональный параметр, запрещает устройству участвовать в процессе выбора DR в NBMA-сетях. По умолчанию при создании ipv6 ospf neighbor соединение устанавливается в режиме eligible.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf neighbor fc00::2
```

ipv6 ospf network

Данная команда определяет тип сети.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 ospf network <TYPE>
no ipv6 ospf network
```

Параметры

<TYPE> – тип сети:

- broadcast – тип соединения широковещательный;
- non-broadcast – тип соединения NBMA;
- point-to-multipoint – тип соединения точка-многоточие;
- point-to-multipoint non-broadcast – тип соединения NBMA точка-многоточие;
- point-to-point – тип соединения точка-точка.

Значение по умолчанию

broadcast

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf network point-to-point
```

ipv6 ospf passive-interface

Данная команда переводит интерфейс в пассивный режим, что означает что с данного интерфейса не будут отправляться hello-пакеты а также не будут устанавливаться отношения смежности (соседства), однако подсеть подключенная к такому интерфейсу будет анонсироваться другим участникам процесса. Использование отрицательной формы команды (no) возвращает интерфейс в режим работы с рассылкой hello-пакетов и установкой соседства.

Синтаксис

```
[no] ipv6 ospf passive-interface
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-twe)# ipv6 ospf passive-interface
```

ipv6 ospf poll-interval

Данная команда устанавливает интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить hello-пакет соседу, даже в случае, если сосед неактивен.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ipv6 ospf poll-interval <TIME>
no ipv6 ospf poll-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1..255].

Значение по умолчанию

120 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf poll-interval 60
```

ipv6 ospf priority

Данной командой устанавливается приоритет маршрутизатора, который используется для выбора DR и BDR.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 ospf priority <VALUE>
no ipv6 ospf priority
```

Параметры

<VALUE> – приоритет интерфейса, принимает значения [0..255].

Значение по умолчанию

120

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf priority 300
```

ipv6 ospf retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ipv6 ospf retransmit-interval <TIME>
no ipv6 ospf retransmit-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)#ipv6 ospf restransmit-interval 4
```

ipv6 ospf ttl-security-hops

Данной командой определяется значение параметра ttl в исходящих ospf-пакетах. Также во всех входящих ospf-пакетах будет осуществляться проверка, что значение ttl больше либо равно '255 - ttl-security-hops'.

Использование отрицательной формы команды (no) устанавливает значение ttl-security-hops по умолчанию.

Синтаксис

```
ipv6 ospf ttl-security-hops <VALUE>
no ipv6 ospf ttl-security-hops
```

Параметры

<VALUE> – значение ttl [1..255].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)#ipv6 ospf ttl-security-hops 200
```

ipv6 ospf wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
ipv6 ospf wait-interval <TIME>
no ipv6 ospf wait-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

40 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 ospf  wait-interval 60
```

ipv6 router ospf

Командой добавляется OSPFv3-процесс в систему и осуществляется переход в режим настройки параметров OSPFv3-процесса.

Использование отрицательной формы команды (no) удаляет OSPFv3-процесс из системы.

Синтаксис

```
[no] ipv6 router ospf <ID> [vrf <VRF>]
```

Параметры

<ID> – номер автономной системы процесса, принимает значения [1..65535].

<VRF> – имя экземпляра VRF, в котором будет работать OSPFv3-процесс, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 router ospf 300
rtt(config-ipv6-ospf)#
```

ipv6 router ospf log-adjacency-changes

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPFv3.

Использование отрицательной формы команды (no) отключает вывод информации.

Синтаксис

```
[no] ipv6 router ospf log-adjacency-changes
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 router ospf log-adjacency-changes
```

maximum-paths

Данной командой определяется максимальное количество равноценных маршрутов до цели.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
maximum-paths <PATHS>
no maximum-paths
```

Параметры

<PATHS> – количество допустимых равноценных маршрутов до цели, принимает значения [1..32].

Значение по умолчанию

16

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример 1

```
rtt(config-ospf)# maximum-paths 10
```

Пример 2

```
rtt(config-ipv6-ospf)# maximum-paths 10
```

opaque-lsa disable

Данной командой отключается поддержка OPAQUE LSA сообщений для текущего процесса OSPF. В результате в процессе установки смежности маршрутизатор сообщит об отсутствии поддержки всех типов OPAQUE LSA.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

После отключения поддержки OPAQUE LSA функционал, использующий данный тип LSA (например, Graceful-restart), будет деактивирован.

Синтаксис

```
[no] opaque-lsa disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Поддержка OPAQUE LSA включена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

Пример

```
rtt(config-ospf)# opaque-lsa disable
```

preference

Данная команда определяет приоритетность маршрутов процесса OSPF/OSPFv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
preference <VALUE>
no preference
```

Параметры

<VALUE> – приоритетность маршрутов процесса OSPF, принимает значения в диапазоне [1..255].

Значение по умолчанию

Наследуется от глобального значения preference, устанавливаемого командой ip protocols preference.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример 1

```
rtt(config-ospf)# preference 30
```

Пример 2

```
rtt(config-ipv6-ospf)# preference 30
```

retransmit-interval

Данной командой устанавливается интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, который не получил подтверждение о получении (например, Database Description пакет или Link State Request пакеты).

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
retransmit-interval <TIME>
no retransmit-interval
```

Параметры

<TIME> – время в секундах, принимает значения [2..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Пример

```
rtt(config-ospf-vlink)# restransmit-interval 4
```

router ospf

Данной командой добавляется OSPF-процесс в систему и осуществляется переход в режим настройки параметров OSPF-процесса.

Использование отрицательной формы команды (no) удаляет OSPF-процесс из системы.

Синтаксис

```
[no] router ospf <ID> [vrf <VRF>]
```

Параметры

<ID> – номер автономной системы процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router ospf 300
rtt(config-ospf)#
```

router ospf log-adjacency-changes

Данная команда позволяет включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPFv2.

Использование отрицательной формы команды (no) отключает вывод информации.

Синтаксис

```
[no] router ospf log-adjacency-changes
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router bgp log-adjacency-changes
```

router-id

Данной командой устанавливается идентификатор маршрутизатора.

Использование отрицательной формы команды (no) удаляет идентификатор.

Синтаксис

```
router-id { <ID> | <IF> | <TUN> }
no router-id
```

Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF

CONFIG-OSPFV3

Пример

```
rtt(config-ospf)# router-id 1.1.1.1
```

show ip ospf

Данная команда отображает таблицу маршрутизации OSPF, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

Синтаксис

```
show ip ospf [ <ID> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, для которого будет отображена таблица маршрутизации OSPF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

Отображение таблицы маршрутизации:

```
rtt# show ip ospf
O       2.2.2.0/24      [150/10] dev gi1/0/1    [ospf2 19:40:31]   (2.2.2.2)
```

show ip ospf database

Данная команда отображает базу данных OSPF.

Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] database [ active ] [ area-id <AREA-ID> ] [
detailed ] [ { asbr-summary | external | network | nssa-external | opaque-link |
router | summary } ] [ lsa-id <LSA-ID> ] [ router-id <ROUTER-ID> ] [ self-originate
] [ statistic ]
```

Параметры

active – показать LSA, время жизни которых менее 3600 секунд;

<AREA-ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

asbr-summary – показать LSA Type 4 - ASBR-summary-LSA;

detailed – показать расширенную информацию о LSA;

external – показать LSA Type 5 - AS-external-LSA;

<LSA-ID> – идентификатор LSA, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

network – показать LSA Type 2 - Network-LSA;

nssa-external – показать LSA Type 7 - NSSA-external-LSA;

opaque-link – показать Opaque LSA;

router – показать LSA Type 1 - Router-LSA;

<ROUTER-ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

self-originate – показать локально сгенерированные LSA;

statistic – показать статистическую информацию о LSA, хранящихся в LSDB;

summary – показать LSA Type 3 - Summary-LSA;

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip ospf 111 dababase
Global
 Type    LS ID           Router           Age  Sequence  Checksum
 0005  10.166.11.12    10.1.0.1         1020  80000013    01b7
 0005  0.0.0.0         10.166.11.1       245  80000010    aa48
 0005  10.62.19.128    10.166.11.1       725  8000000e    6d2b
 0005  10.62.20.0      10.166.11.1       731  8000000d    69af
 0005  10.62.20.128    10.166.11.1       244  80000010    5e37
 0005  10.62.21.128    10.166.11.1       244  80000010    5341
 0005  10.166.11.0     10.166.11.1       245  80000010    cc6d
 0005  10.166.11.12    10.166.11.1       245  80000010    54d9
Area 0.0.11.1
 Type    LS ID           Router           Age  Sequence  Checksum
 0001  10.1.0.1        10.1.0.1         1015  80000067    989e
 0001  10.166.11.1     10.166.11.1      1021  80000018    8d96
 0002  10.166.11.14    10.166.11.1      1021  80000001    68a5
```

show ip ospf interface

Данная команда отображает информацию об OSPF-интерфейсе.

Синтаксис

```
show ip ospf interface [  vrf <VRF> ] [ <IF> | <TUN> ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация об OSPF-интерфейсе, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip ospf interface gi1/0/1
Interface:                    gigabitethernet 1/0/1
Internet Address:             25.25.0.1/24
Router ID:                    6.0.0.1
Network Type:                 broadcast
Area:                         0.0.0.0 (0)
Interface has:                no authentication
```

```
Transmit:                     1
State:                        dr
Priority:                     128
Cost:                         10
ECMP weight:                  1
Hello timer:                  10
Wait timer:                   40
Dead timer:                   40
Retransmit timer:             5
Designed router (ID):         6.0.0.1
Designed router (IP):         25.25.0.1
Backup designed router (ID):  6.0.0.3
Backup designed router (IP):  25.25.0.3
Neighbor Count:               0
Adjacent neighbor count:      0
```

show ip ospf neighbors

Данная команда отображает информацию о всех соседях или соседях определенного OSPF-процесса.

Синтаксис

```
show ip ospf [ <ID> [ vrf <VRF> ] ] neighbors
show ip ospf neighbors[ <ID> [ vrf <VRF> ] ]
```

Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip ospf neighbors
Router ID        Pri          State       DTime    Interface  Router IP
160.0.0.2          0         full/ptp     00:53    vlink0     160.0.0.2
95.0.0.1           1         full/dr      00:31    gi1_15     115.0.0.10
10.100.100.2     128         full/ptp     00:37    gre_25     25.25.0.2
153.0.0.1          1         full/bdr     00:30    po1        1.1.0.2
10.100.100.2     128         2way/other   00:34    gi1_14.25  14.25.0.2
24.24.24.24       15         full/bdr     00:32    te1_1      24.0.0.2
```

show ip ospf virtual-links

Данная команда отображает информацию о виртуальных соединениях.

Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] virtual-links
```

Параметры

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: 160.0.0.2
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

show ipv6 ospf

Данная команда отображает таблицу маршрутизации OSPFv3, если не указан аргумент. При указании процесса выводит информацию о конфигурации интерфейсов по данному процессу.

Синтаксис

```
show ip ospf <ID> [ vrf <VRF> ] database [ active ] [ area-id <AREA-ID> ] [
detailed ] [ { asbr-summary | external | link-local | network | nssa-external |
opaque-link | prefix | router | summary } ] [ lsa-id <LSA-ID> ] [ router-id
<ROUTER-ID> ] [ self-originate ] [ statistic ]
```

Параметры

active – показать LSA, время жизни которых менее 3600 секунд;

<AREA-ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

asbr-summary – показать LSA Type 0x2004 - Inter-Area-Router-LSA;

detailed – показать расширенную информацию о LSA;

external – показать LSA Type 0x4005 - AS-external-LSA;

link-local - показать LSA Type 0x0008 - Link-LSA;

<LSA-ID> – идентификатор LSA, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

network – показать LSA Type 0x2002 - Network-LSA;

nssa-external – показать LSA Type 0x2007 - NSSA-LSA;

opaque-link – показать Opaque LSA;

prefix – показать LSA Type 0x2009 - Intra-Area-Prefix-LSA;

router – показать LSA Type 0x2001 - Router-LSA;

<ROUTER-ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

self-originate – показать локально сгенерированные LSA;

statistic – показать статистическую информацию о LSA, хранящихся в LSDB;

summary – показать LSA Type 0x2003 - Inter-Area-Prefix-LSA;

<ID> – номер OSPF-процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPF-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

Отображение таблицы маршрутизации:

```
rtt# show ipv6 ospf
O       fc00::/120      [150/10] dev gi1/0/5    [ospf2 19:39:18]   (2.2.2.2)
```

show ipv6 ospf database

Данная команда отображает таблицу данных OSPFv3.

Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] database
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена таблица данных, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 ospf 111 dababase
Global
 Type    LS ID           Router           Age  Sequence  Checksum
 0005  10.166.11.12    10.1.0.1         1020  80000013    01b7
 0005  0.0.0.0         10.166.11.1       245  80000010    aa48
 0005  10.62.19.128    10.166.11.1       725  8000000e    6d2b
 0005  10.62.20.0      10.166.11.1       731  8000000d    69af
 0005  10.62.20.128    10.166.11.1       244  80000010    5e37
 0005  10.62.21.128    10.166.11.1       244  80000010    5341
 0005  10.166.11.0     10.166.11.1       245  80000010    cc6d
 0005  10.166.11.12    10.166.11.1       245  80000010    54d9
Area 0.0.11.1
 Type    LS ID           Router           Age  Sequence  Checksum
 0001  10.1.0.1        10.1.0.1         1015  80000067    989e
 0001  10.166.11.1     10.166.11.1      1021  80000018    8d96
 0002  10.166.11.14    10.166.11.1      1021  80000001    68a5
```

show ipv6 ospf interface

Данная команда отображает информацию об интерфейсах, на которых включен протокол OSPFv3.

Синтаксис

```
show ipv6 ospf interface [ vrf <VRF> ] [ <IF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация об OSPFv3-интерфейсе, задается строкой до 31 символа;

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 ospf interface gigabitethernet 1/0/14
rtt# Interface:                    gigabitethernet 1/0/14
Internet Address:             IID 0
Router ID:                    88.88.88.88
Network Type:                 broadcast
Area:                         0.0.0.0 (0)
Transmit:                     1
State:                        backup
Priority:                     128
Cost:                         10
ECMP weight:                  1
Hello timer:                  10
Wait timer:                   40
Dead timer:                   40
Retransmit timer:             5
Designed router (ID):         77.0.0.1
Designed router (IP):         fe80::c602:46ff:feed:0
Backup designed router (ID):  88.88.88.88
Backup designed router (IP):  fe80::1:2ff:fe03:463
Neighbor Count:               1
Adjacent neighbor count:      1
Adjacent with neighbor:    77.0.0.1 (dr)
```

show ipv6 ospf neighbors

Данная команда отображает информацию о всех соседях или соседях определенного OSPFv3-процесса.

Синтаксис

```
show ipv6 ospf [<ID> [vrf <VRF>]] neighbors
```

Параметры

<ID> – номер процесса, принимает значения [1..65535], опциональный параметр.

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация по соседям, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 ospf neighbor
Router ID          Pri      State      DTime   Interface  Router IP
77.0.0.1             1     full/dr     00:32   gi1_14     fe80::c602:46ff:feed:0
33.33.33.33        128     full/bdr    00:35   gi1_18     fe80::20:3ff:fea0:498
```

show ipv6 ospf virtual-links

Данная команда отображает информацию о виртуальных соединениях.

Синтаксис

```
show ipv6 ospf <ID> [vrf <VRF>] virtual-links
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF для OSPFv3-процесса, в котором будет отображена информация по виртуальным соединениям, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 ospf 10 virtual-links
Virtual Link to router 160.0.0.2 is ptp
Peer IP: fe80::20:3ff:fea0:498
Transit area: 1.1.1.1
Interface has no authentication
Timer intervals configured: Hello 10, Dead 60, Retransmit 5, Wait 60
Adjacency State full
```

summary-address

Командой включается суммаризация или скрытие подсетей. Суммаризация или скрытие подсетей при помощи данной команды возможны только для межзоновых маршрутов (типа IA).

Использование отрицательной формы команды (no) отключает суммаризацию или скрытие подсетей.

Синтаксис

```
summary-address { <ADDR/LEN> | <IPV6-ADDR/LEN> } { advertise | not-advertise }
no summary-address <ADDR/LEN>
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

advertise – при указании команды вместо указанных подсетей будет анонсироваться суммарная подсеть;

not-advertise – при указании команды подсети, входящие в указанную подсеть, анонсироваться не будут.

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример

```
rtt(config-ospf-area)# summary-address 192.168.16.0/24
```

virtual-link

Данной командой устанавливается виртуальное соединение между основной и удаленными областями, имеющими несколько областей между ними.

Использование отрицательной формы команды (no) удаляет созданное виртуальное соединение.

Синтаксис

```
[no] virtual-link <ID>
```

Параметры

<ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-AREA

CONFIG-OSPFV3-AREA

Пример

```
rtt(config-ospf-area)# virtual-link 160.0.0.2
```

wait-interval

Данной командой определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
wait-interval <TIME>
no wait-interval
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535].

Значение по умолчанию

40

Необходимый уровень привилегий

10

Командный режим

CONFIG-OSPF-VLINK

CONFIG-OSPFV3-VLINK

Пример

```
rtt(config-ospf-vlink)# ospf wait-interval 60
```

21.10. Настройка протокола IS-IS

address-family

Данная команда включает обмен маршрутами указанного семейства по протоколу IS-IS.
Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
[no] address-family { ipv4 | ipv6 }
```

Параметры

- ipv4 – семейство адресов IPv4;
- ipv6 – семейство адресов IPv6.

Значение по умолчанию

ipv4 – включено;

ipv6 – отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# address-family ipv4
```

authentication area algorithm

Данная команда позволяет выбрать алгоритм аутентификации для L1-уровня, шифроваться будут LSP/SNP-пакеты.

Использование отрицательной формы команды (no) устанавливает значение временного интервала по умолчанию.

Синтаксис

```
authentication area algorithm <ALGORITHM>
no authentication area algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication area algorithm cleartext
```

authentication area key

Данная команда устанавливает пароль аутентификации для L1-уровня, этим паролем будут шифроваться LSP/SNP-пакеты.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication area key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication area key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой, длинной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication area key ascii-text password
```

authentication area key chain

Данная команда устанавливает список ключей для аутентификации для L1-уровня.

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

Синтаксис

```
authentication area key chain <KEYCHAIN>
no authentication area key chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication area key chain key-1
```

authentication domain algorithm

Данная команда устанавливает алгоритм аутентификации для L2-уровня, шифроваться будут LSP/SNP-пакеты.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
authentication domain algorithm <ALGORITHM>
no authentication domain algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication domain algorithm cleartext
```

authentication domain key

Данная команда устанавливает пароль аутентификации для L2-уровня, этим паролем будут шифроваться LSP/SNP-пакеты.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication domain key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication domain key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой, длинной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication domain key ascii-text password
```

authentication domain key chain

Данная команда устанавливает список ключей для аутентификации для L2-уровня.

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

Синтаксис

```
authentication domain key chain <KEYCHAIN>
no authentication domain key chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# authentication domain key chain key-1
```

clear isis

Данная команда перезапускает определенный IS-IS процесс. Команда без параметров перезапускает все процессы IS-IS, настроенные на маршрутизаторе.

Синтаксис

```
clear isis [ <ID> ] [ vrf <VRF_NAME> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear isis 15
```

hostname dynamic

Данная команда включает использование TLV 137, которое позволяет передавать hostname маршрутизатора в LSP.

Использование отрицательной формы команды (no) отключает данную возможность.

Синтаксис

```
[no] hostname dynamic
```

Параметры

Команда не содержит параметров

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# hostname dynamic
```

is-type

Данная команда определяет, на каком уровне маршрутизации будет работать текущий процесс IS-IS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
is-type <LEVEL>
no is-type
```

Параметры

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-1-2 – работа производится и на 1, и на 2 уровне;
- level-2-only – работа производится только на 2 уровне.

Значение по умолчанию

level-1-2

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# is-type level-1
```

isis adjacency point-to-point mode

Данная команда устанавливает способ установления соседства в режиме "точка-точка".

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
isis adjacency point-to-point mode <MODE>
no isis adjacency point-to-point mode
```

Параметры

<MODE> – режим установления соседства:

- two-way-only – IIH с TLV 240 "Point-to-Point Three-Way Adjacency" будут обрабатываться, но сам TLV 240 игнорируется;
- three-way-only – IIH без TLV 240 обрабатываться не будут;
- two-way-three-way – возможность работы как с TLV 240 так и без.

Значение по умолчанию

Разрешены оба способа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis adjacency point-to-point mode three-way-only
```

isis authentication algorithm

Данная команда устанавливает алгоритм аутентификации для hello-пакетов (IIH), выходящих с данного интерфейса.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
isis authentication algorithm <ALGORITHM> [ <LEVEL> ]
no isis authentication algorithm [ <LEVEL> ]
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis authentication algorithm cleartext level-1
```

isis authentication key

Данная команда устанавливает пароль для аутентификации hello-пакетов (IIH).

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
isis authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> } [
<LEVEL> ]
no isis authentication key [ <LEVEL> ]
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой длиной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis authentication key ascii-text password level-1
```

isis authentication key chain

Данная команда устанавливает список ключей для аутентификации hello-пакетов (IIH).

Использование отрицательной формы команды (no) удаляет привязку к списку ключей.

Синтаксис

```
isis authentication key chain <KEYCHAIN> [ <LEVEL> ]
no isis authentication key chain [ <LEVEL> ]
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой [1..16] символов.

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis authentication key chain key-1 level-1
```

isis bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола IS-IS.

Использование отрицательной формы команды (no) отключает на интерфейсе протокол BFD для протокола IS-IS.

Синтаксис

```
[no] isis bfd-enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Протокол BFD для протокола IS-IS отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

Пример

```
rtt(config-lt)# isis bfd-enable
```

isis circuit-type

Данная команда определяет, на каком уровне маршрутизации будет работать текущий процесс IS-IS на конкретном интерфейсе.

Использование отрицательной формы команды (no) устанавливает уровень работы протокола IS-IS на интерфейсе в соответствии с конфигурацией самого IS-IS процесса.

Синтаксис

```
isis circuit-type <LEVEL>
no isis circuit-type
```

Параметры

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-1-2 – работа производится и на 1, и на 2 уровне;
- level-2-only – работа производится только на 2 уровне.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis circuit-type level-1
```

isis csnp-interval

Данная команда устанавливает интервал генерации и отправки Complete Sequence Number Protocol Data Units (CSNP PDUs).

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis csnp-interval <TIME> [ <LEVEL> ]
no isis csnp-interval [ <LEVEL> ]
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis csnp-interval 33 level-1
```

isis enable

Данная команда включает работу протокола IS-IS на интерфейсе.

Использование отрицательной формы команды (no) выключает IS-IS процесс.

Синтаксис

```
isis enable
no isis enable
```

Параметры

Команда не содержит параметров

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis enable
```

isis hello-interval

Данная команда устанавливает интервал отправки hello-пакетов (IIH) с интерфейса маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis hello-interval <TIME> [ <LEVEL> ]
no isis hello-interval [ <LEVEL> ]
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis hello-interval 33 level-1
```

isis hello-multiplier

Данная команда устанавливает множитель для вычисления и отправки Hold Time (Hold Time = hello-timer * hello-multiplier) в hello пакетах (IIH).

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis hello-multiplier <VALUE> [ <LEVEL> ]
no isis hello-multiplier [ <LEVEL> ]
```

Параметры

<VALUE> – число, принимающее значения [3..1000];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

3

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis hello-multiplier 34 level-1
```

isis instance

Данная команда определяет принадлежность интерфейса к определенному IS-IS процессу.

Использование отрицательной формы команды (no) удаляет принадлежность интерфейса к IS-IS процессу.

Синтаксис

```
isis instance <ID>
no isis instance
```

Параметры

<ID> – номер процесса, принимает значения [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis instance 35
```

isis ipv6-bfd-enable

Данной командой на интерфейсе включается протокол BFD для протокола ISIS (для address-family IPv6).

Использование отрицательной формы команды (no), на интерфейсе отключается протокол BFD для протокола ISIS.

Синтаксис

```
[no] isis ipv6-bfd-enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Протокол BFD для протокола ISIS отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis ipv6-bfd-enable
```

isis lsp-interval

Данная команда задает интервал между успешными последовательными передачами Link-state Packets (LSPs) в Broadcast-сети.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis lsp-interval <TIME> [ <LEVEL> ]
no isis lsp-interval [ <LEVEL> ]
```

Параметры

<TIME> – время в миллисекундах, принимает значения [1-10000];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

100

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis lsp-interval 36 level-2
```

isis lsp-retransmit-interval

Данная команда задает интервал повторного распространения всех Link-state Packets (LSPs) в PtP-сети.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis lsp-retransmit-interval <TIME> [ <LEVEL> ]
no isis lsp-retransmit-interval [ <LEVEL> ]
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis lsp-retransmit-interval 37 level-2
```

isis metric

Данная команда устанавливает определенное значение метрики для интерфейса.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis metric <VALUE> [<LEVEL>]
no isis metric [<LEVEL>]
```

Параметры

<VALUE> – число, принимающее значения [1..16777215];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis metric 38 level-1
```

isis network point-to-point

Данная команда переводит текущий интерфейс в режим работы point-to-point протокола IS-IS.

Использование отрицательной формы команды (no) прекращает работу в данном режиме.

Синтаксис

```
[no] isis network point-to-point
```

Параметры

Команда не содержит параметров

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-gre)# isis network point-to-point
```

isis priority

Данная команда устанавливает приоритет при выборе DIS.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis priority <VALUE> [ <LEVEL> ]
no isis priority [ <LEVEL> ]
```

Параметры

<VALUE> – число, принимающее значения [0..127];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

64

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis priority 39 level-2
```

isis psnp-interval

Данная команда устанавливает интервал генерации и отправки Partial Sequence Number Protocol Data Units (PSNP PDUs).

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
isis psnp-interval <TIME> [ <LEVEL> ]
no isis psnp-interval [ <LEVEL> ]
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

2

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis psnp-interval 40 level-1
```

lsp-refresh-interval

Данная команда устанавливает интервалы обновления/генерации собственных LSP.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
lsp-refresh-interval { min | max } <TIME> [ <LEVEL> ]
no lsp-refresh-interval { min | max } [ <LEVEL> ]
```

Параметры

min – минимальный интервал обновления/генерации;

max – максимальный интервал обновления/генерации;

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

min 30

max 900

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# lsp-refresh-interval min 41 level-1
```

max-lsp-lifetime

Данная команда задает время жизни собственных LSP.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
max-lsp-lifetime <TIME> [ <LEVEL> ]
no max-lsp-lifetime [ <LEVEL> ]
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

1200

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# max-lsp-lifetime 42 level-2
```

metric-style

Данная команда устанавливает тип метрики, который будет использоваться в работе IS-IS процесса.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
metric-style { narrow | wide | transition } [ <LEVEL> ]
no metric-style [ <LEVEL> ]
```

Параметры

narrow – информация о достижимости сетей распространяется с помощью TLV 128;

wide – информация о достижимости сетей распространяется с помощью TLV 135;

transition – информация о достижимости сетей распространяется обоими способами;

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

narrow

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# metric-style wide level-1
```

net

Данная команда устанавливает NET (Network Entity Title) адрес. Он репрезентует AFI (Authority and Format Identifier), Area ID (зону к которой принадлежит маршрутизатор), System ID (идентификатор маршрутизатора) и значение Selector.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
[no] net <NET>
```

Параметры

<NET> – NET-адрес, формат: ff[.ffff.ffff.ffff.ffff.ffff.ffff].ffff.ffff.ffff.00, где f – может принимать значения от [1..F].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# net 49.0123.1010.1010.1010.00
```

no isis hello-padding

Данная команда задает отключает режим использования TLV 8 в hello-пакетах (IIH).

Использование положительной формы команды (без ключа no) устанавливает значению по умолчанию.

Синтаксис

```
[no] isis hello-padding
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-LOOPBACK

CONFIG-LT

Пример

```
rtt(config-if-gi)# isis hello-padding
```

preference

Данная команда устанавливает приоритетность маршрутов данного IS-IS процесса для основной таблицы маршрутизации.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
preference <VALUE>
no preference
```

Параметры

<VALUE> – принимает значения [1..255].

Значение по умолчанию

160

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# preference 43
```

router isis

Данная команда создаёт IS-IS процесс и осуществляет переход в режим настройки параметров этого процесса.

Использование отрицательной формы команды (no) удаляет IS-IS процесс из системы.

Синтаксис

```
[no] router isis <ID> [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# router isis 44
```

show ip isis

Данная команда выводит внутреннюю таблицу маршрутизации.

Синтаксис

```
show ip isis [ <ID> ] [ <ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<ADDR/LEN> – адрес, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LEVEL> – уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip isis 45
IPv4 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 06:51:02.110
Address                       Type Metric Via
3.3.3.11/32                   L1   20     2.2.2.11 on gi1/0/2

IPv4 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 06:51:12.440
Address                       Type Metric Via
3.3.3.22/32                   IA   20     1.1.1.2 on gi1/0/1
```

show ipv6 isis

Данная команда выводит внутреннюю ipv6-таблицу маршрутизации.

Синтаксис

```
show ipv6 isis [ <ID> ] [ <IPV6-ADDR/LEN> ] [ <LEVEL> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<IPV6-ADDR/LEN> – IPv6-адрес подсети с маской, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];

<LEVEL> — уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 isis 46
IPv6 local Level 1 RIB for IS-IS process
Last update: 2020-01-24 07:08:37.761
Address                       Type Metric Via
6666::/64                     L1   20     fe80::aaf9:4bff:feab:813a on gi1/0/2

IPv6 local Level 2 RIB for IS-IS process
Last update: 2020-01-24 07:07:22.917
Address                       Type Metric Via
2001::/64                     IA   20     fe80::aaf9:4bff:feac:b4a1 on gi1/0/1
```

show isis database

Данная команда выводит информацию по локальной базе всех полученных и сгенерированных LSP (LSDB).

Синтаксис

```
show isis database [ <ID> ] [ <LEVEL> ] [ <LSP-ID> ] [ detailed ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> — уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

<LSP-ID> – ID LSP пакета, задаётся в виде xxxx.xxxx.xxxx.yy-zz, где на месте x – System-ID, y – Pseudonode-ID, z – номер фрагмента. Все числа могут принимать значения [0..F];

detailed – развёрнутая информация о выбранном пункте;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show isis database 47
IS-IS Level 1 Link State Database
LSPID                 Hostname      LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * rtt           0x0000000f   0x5eba        960           1/0/0
1111.1111.1111.00-00   rtt           0x00000010   0x4402        1071          0/0/0
1111.1111.1111.01-00   rtt           0x00000004   0x1a99        985           0/0/0

IS-IS Level 2 Link State Database
LSPID                 Hostname      LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
1010.1010.1010.00-00 * rtt           0x00000010   0x7776        1082          0/0/0
1222.1222.1222.01-00   rtt           0x00000003   0xb87a        997           0/0/0
1222.1222.1222.00-00   rtt           0x0000000e   0x84c4        997           0/0/0
```

show isis hostname

Данная команда выводит соответствие System-id и имени роутера.

Синтаксис

```
show isis hostname [ <ID> ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show isis hostname 48
IS-IS 12
Level  System ID        Dynamic Hostname
   1   1010.1010.1010 * rtt
   1   1111.1111.1111   rtt
   2   1010.1010.1010 * rtt
   2   1222.1222.1222   rtt
```

show isis neighbors

Данная команда отображает информацию по состоянию IS-IS соседств.

Синтаксис

```
show isis neighbors [ <ID> ] [ <LEVEL> ] [ detailed ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> — уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

detailed – развёрнутая информация о выбранном пункте;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show isis neighbors 49 detailed
IS-IS Level 1 Neighbors
System ID       Hostname      Interface         State         Holdtime  SNPA
1111.1111.1111  rtt           gi1/0/2           Up            10        a8f9.4bab.813a

IS-IS Level 2 Neighbors
System ID       Hostname      Interface         State         Holdtime  SNPA
1222.1222.1222  rtt           gi1/0/1           Up            10        a8f9.4bac.b4a1
```

show isis topology

Данная команда выводит полную таблицу достижимости всех сетей и узлов.

Синтаксис

```
show isis topology [ <ID> ] [ <LEVEL> ] [ ipv4 | ipv6 ] [ vrf <VRF> ]
```

Параметры

<ID> – номер процесса, принимает значения [1..65535];

<LEVEL> — уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне;

ipv4 – семейство адресов IPv4;

ipv6 – семейство адресов IPv6;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show isis topology 50 level-2 Ipv4
IS-IS path to Level 2 routers (narrow metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                        Type             Metric   Next-Hop           Interface
1.1.1.0/24                    IPv4 Internal    0        --

IS-IS path to Level 2 routers (wide metric-style)
Last update: 2020-01-24 07:08:47.783
Vertex                        Type             Metric   Next-Hop           Interface
3.3.3.22/32                   IPv4 IA          20       rtt                gi1/0/1
1.1.1.0/24                    IPv4             0        --
```

spf-timeout

Данная команда устанавливает таймаут перед следующим расчётом SPF.

Использование отрицательной формы команды (no) устанавливает значению по умолчанию.

Синтаксис

```
spf-timeout <TIME> [ <LEVEL> ]
no spf-timeout [ <LEVEL> ]
```

Параметры

<TIME> — время в миллисекундах, принимает значения [1..10000];

<LEVEL> — уровень работы протокола IS-IS:

- level-1 – работа производится только на 1 уровне;
- level-2 – работа производится только на 2 уровне.

Значение по умолчанию

5000

Необходимый уровень привилегий

10

Командный режим

CONFIG-ISIS

Пример

```
rtt(config-isis)# spf-timeout 51 level-1
```

22. НАСТРОЙКИ MPLS

22.1. Общие настройки MPLS

forwarding interface

Данной командой включается передачу MPLS-пакетов на интерфейсе.

Использование отрицательной формы команды (no) отключает передачу MPLS-пакетов на интерфейсе.

Синтаксис

```
[no] forwarding interface { <IF> | <TUN> }
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-MPLS

Пример

```
rtt(config-mpls)# forwarding interface gi 1/0/14
```

history statistics

Данной командой включается запись статистики использования псевдопровода/динамического псевдопровода.

Использование отрицательной формы команды (no) отключает запись статистики использования псевдопровода/динамического псевдопровода.

Синтаксис

```
[no] history statistics
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

CONFIG-L2VPN-PW

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# history statistics
```

l2vpn

Данной командой осуществляется переход в контекст настройки MPLS-L2-туннелей.

Использование отрицательной формы команды (no) удаляет все настройки MPLS-L2-туннели.

Синтаксис

```
[no] l2vpn
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-MPLS

Пример

```
rtt(config-mpls)# l2vpn
```

ldp

Данной командой осуществляется переход в контекст настройки протокола LDP.

Использование отрицательной формы команды (no) удаляет все настройки протокола LDP.

Синтаксис

```
[no] ldp
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-MPLS

Пример

```
rtt(config-mpls)# ldp
```

mpls

Данной командой осуществляется переход в контекст настройки функционала MPLS.

Использование отрицательной формы команды (no) удаляет все настройки функционала LDP.

Синтаксис

```
[no] mpls
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# mpls
```

system cpu load-balance mpls passenger

Данной командой включается функционал балансировки трафика между CPU маршрутизатора.

Использование отрицательной формы команды (no) отключает функционал балансировки.

Синтаксис

```
system cpu load-balance mpls passenger [ <OPTION> ]
no system cpu load-balance mpls passenger [ <OPTION> ]
```

Параметры

Параметр <OPTION> принимает значения:

- ip – включает поиск и добавление в расчет хеша ip-src и ip-dst;
- ipoe-pw-with-cw – возможность явно указать, что используется заголовок с Control Word.
- ipoe-pw-without-cw – возможность явно указать, что не используется заголовок с Control Word.

Значение по умолчанию

Отключено. В расчете хеш не используются ip-src и ip-dst.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# system cpu load-balance mpls passenger ip
```

22.2. Настройки протокола LDP

address-family

Команда позволяет перейти в режим конфигурации параметров LDP для выбранной address family.

Использование отрицательной формы команды (no) удаляет все параметры из конфигурации.

Синтаксис

```
[no] address-family ipv4
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
rtt(config-ldp)# address-family ipv4
```

advertise-labels

Команда позволяет задать FEC, для которых протокол LDP будет выделять и распространять метки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

Синтаксис

```
advertise-labels <OBJ-GROUP-NETWORK-NAME>
no advertise-labels
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Значение по умолчанию

Метки выделяются и анонсируются для всех FEC.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
rtt(config-ldp)# advertise-labels mpls_adv
```

clear mpls ldp

Данная команда служит для сброса LDP-процесса или для очистки таблицы LDP-соседей.

Синтаксис

```
clear mpls ldp [neighbor <ADDR> ]
```

Параметры

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT1# clear mpls ldp neighbor 1.1.1.1
```

description

Данная команда позволяет добавить описание для LDP-соседа.

Использование отрицательной формы команды (no) удаляет описание LDP-соседа.

Синтаксис

```
description <STRING>
no description
```

Параметры

<STRING> – строка от 1 до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp-neig)# description RTT1
```

discovery hello holdtime

Данная команда устанавливает значение параметра holdtime. Holdtime – интервал времени, в течение которого должно прийти хотя бы одно LDP hello сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
discovery hello holdtime <TIME>
no discovery hello holdtime
```

Параметры

<TIME> – время в секундах в интервале [3..65535].

Значение по умолчанию

15

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

Пример

```
rtt(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

discovery hello interval

Данная команда устанавливает значение параметра hello interval. Hello interval – промежуток времени между отправкой LDP hello сообщения.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
discovery hello interval <TIME>
no discovery hello interval
```

Параметры

<TIME> – время в секундах в интервале [1..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

CONFIG-LDP-AF-IPV4-IF

Пример

```
rtt(config-ldp-af-ipv4-if)# discovery hello interval 15
```

discovery targeted-hello accept

Данная команда разрешает прием targeted-hello сообщений.

Использование отрицательной формы команды (no) запрещает прием targeted-hello сообщений.

Синтаксис

```
discovery targeted-hello accept
no targeted-hello accept
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-AF-IPV4

Пример

```
rtt(config-ldp-af-ipv4)# discovery targeted-hello accept
```

discovery targeted-hello interval

Данная команда устанавливает значение параметра hello interval. Hello interval – промежуток времени между отправкой LDP hello сообщений для targeted-соседа.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
discovery targeted-hello interval <TIME>
no discovery targeted-hello interval
```

Параметры

<TIME> – время в секундах в интервале [1..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp-neig)# discovery targeted-hello interval 15
```

discovery targeted-hello holdtime

Данная команда устанавливает значение параметра holdtime для targeted-соседа. Holdtime – интервал времени, в течение которого должно прийти хотя бы одно LDP hello сообщение.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
discovery targeted-hello holdtime <TIME>
no discovery targeted-hello holdtime
```

Параметры

<TIME> – время в секундах в интервале [3..65535].

Значение по умолчанию

45

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp-af-ipv4-if)# discovery hello holdtime 45
```

egress-label-type

С помощью данной команды устанавливается режим, когда для FEC, у которых маршрутизатор является egress lsr, анонсируется метка explicit-null (0) вместо implicit null (3).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
egress-label-type explicit-null
no egress-label-type
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Анонсируется IMPLICIT NULL (3).

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
 rtt(config-ldp)# egress-label-type explicit-null
```

enable

Данной командой включается процесс LDP.

Использование отрицательной формы команды (no) отключает процесс LDP.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
rtt(config-ldp)# enable
```

interface

Данная команда включает прием и рассылку LDP hello сообщений на интерфейсе для определенной address-family.

Использование отрицательной формы команды (no) отключает прием и рассылку LDP hello сообщений на интерфейсе.

Синтаксис

```
[no] interface { <IF> | <TUN> }
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-AF-IPV4

Пример

```
rtt(config-ldp-af-ipv4)# interface gigabitethernet 1/0/1
```

keepalive

С помощью данной команды устанавливается время ожидания от соседа keepalive-сообщения. В случае истечения таймера сосед считается недоступным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
keepalive <TIMER>
no keepalive
```

Параметры

<TIMER> – время в секундах [3..65535].

Значение по умолчанию

180

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp)# keepalive 120
rtt(config-ldp-neig)# keepalive 160
```

neighbor

Данная команда позволяет создать и перейти в режим конфигурации LDP-targeted соседа.

Использование отрицательной формы команды (no) удаляет targeted-соседа из конфигурации.

Синтаксис

```
[no] neighbor <ADDR>
```

Параметры

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
rtt(config-ldp)# neighbor 2.2.2.2
```

password

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
password { <TEXT> | encrypted <ENCRYPTED-TEXT> }
no password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой длиной [8..16] символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером [8..16] байт ([16..32] символа) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp-neig)# password cleartextpassword
```

prefix-list

Данная команда устанавливает список, описывающий подсети, для которых протокол LDP будет выделять и распространять метки.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

Синтаксис

```
prefix-list <PREFIX-LIST-NAME>
```

```
no prefix-list
```

Параметры

<PREFIX-LIST-NAME> – имя списка, задаётся строкой до 31 символа.

Значение по умолчанию

Метки выделяются и анонсируются для всех FEC.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-AF-IPV4

Пример

```
rtt(config-ldp-af-ipv4)# prefix-list LDP_ALLOCATE
```

router-id

Данной командой устанавливается идентификатор маршрутизатора для LDP-протокола.

Использование отрицательной формы команды (no) удаляет идентификатор маршрутизатора для LDP-протокола.

Синтаксис

```
router-id { <ID> | <IF> | <TUN> }
no router-id
```

Параметры

<ID> – идентификатор маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP

Пример

```
 rtt(config-ldp)# router-id 1.1.1.1
```

show mpls ldp binding

Данная команда позволяет вывести информацию по выделенным меткам.

Синтаксис

```
show mpls ldp binding [ detailed ] [ { <ADDR/LEN> | neighbor <ADDR> | [ local
<LABEL-ID> ] [ remote <LABEL-ID> ] } ]
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LABEL-ID> – метка, задается в виде числа в диапазоне [0-104875];

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

detailed – информация выдается в детализированном формате.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT1# sh mpls ldp bindings
10.10.0.4/32:
    local label:    16
    remote label:   16        lsr: 10.10.0.1
    remote label:   149       lsr: 10.10.0.3
    remote label:   imp-null  lsr: 10.10.0.4       inuse
10.10.0.2/32:
    local label:    imp-null
    remote label:   34        lsr: 10.10.0.1
    remote label:   152       lsr: 10.10.0.3
    remote label:   142       lsr: 10.10.0.4
10.10.0.1/32:
    local label:    20
    remote label:   imp-null  lsr: 10.10.0.1       inuse
    remote label:   139       lsr: 10.10.0.3
    remote label:   127       lsr: 10.10.0.4
```

show mpls ldp discovery

Данная команда позволяет вывести оперативную информацию по состоянию процесса обнаружения LDP.

Синтаксис

```
show mpls ldp discovery [ detailed ]
```

Параметры

detailed – информация выдается в детализированном формате.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT1# show mpls ldp discovery detailed
Local LDP ID: 1.1.1.1
Discovery sources:
Interfaces:
gigabitethernet 1/0/1:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 10.10.10.2
Transport IP address: 4.4.4.4
Hold time: 15 seconds
Proposed hold time: 15/15 (local/peer) seconds
Targeted hellos:
1.1.1.1 -> 4.4.4.4:
Hello interval: 5 seconds
Transport IP address: 1.1.1.1
LDP ID: 4.4.4.4
Source IP address: 4.4.4.4
Transport IP address: 4.4.4.4
Hold time: 45 seconds
Proposed hold time: 45/45 (local/peer) seconds
```

show mpls forwarding-table

Данная команда позволяет вывести таблицу коммутации по меткам (LFIB).

Синтаксис

```
show mpls forwarding-table [ { <ADDR/LEN> | label <LABLE-ID> | nexthop <ADDR> |
tunnel <ID> } ]
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<LABEL-ID> – метка, задается в виде числа в диапазоне [0..104875];

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ID> – идентификатор в виде числа в диапазоне [1..2000000].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT1# show mpls forwarding-table
Local     Outgoing                Prefix               Outgoing        Next Hop
label      label           or tunnel ID Interface
-------- -------- ---------------------------------- ---------------- --------------
18         16                   PW   ID 100              --          4.4.4.4
17       imp-null                4.4.4.4/32            gi1/0/1      10.10.10.2
```

show mpls ldp neighbor

Данная команда отображает информацию обо всех LDP-соседях.

Синтаксис

```
show mpls ldp neighbor [ { <ADDR> | interface { <IF> | <TUN> } } ]
```

Параметры

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
RTT1# show mpls ldp neighbor
Peer LDP ID: 4.4.4.4; Local LDP ID 1.1.1.1
State: Operational
TCP connection: 4.4.4.4:38759 - 1.1.1.1:646
Messages sent/received: 195/194
Uptime: 03:07:33
LDP discovery sources:
gigabitethernet 1/0/1
1.1.1.1 -> 4.4.4.4
```

targeted

Данная команда включает отправку targeted hello сообщений.

Использование отрицательной формы команды (no) отключает отправку targeted hello сообщений.

Синтаксис

```
[no] targeted
```

Параметры

Команда не имеет параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-NEIGH

Пример

```
rtt(config-ldp-neig)# targeted
```

transport-address

С помощью данной команды определяется IP-адрес источника для LDP-сообщений, необходимых для установления сессии.

Использование отрицательной формы команды (no) удаляет сконфигурированный IP-адрес источника для LDP-сообщений.

Синтаксис

```
transport-address <ADDR>
no transport-address
```

Параметры

<ADDR> – задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-LDP-AF-IPV4

Пример

```
rtt(config-ldp-af-ipv4)# transport-address 1.1.1.1
```

22.3. Настройки MPLS L2VPN

autodiscovery bgp

Команда позволяет перейти в контекст настройки BGP auto-discovery and singnaling.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

Синтаксис

```
[no] autodiscovery bgp
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-vpls)# autodiscovery bgp
```

bridge-group

С помощью данной команды задается VSI (Virtual Switching Instance).

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

Синтаксис

```
bridge-group <INDEX>
no bridge-group
```

Параметры

<INDEX> – индекс bridge-group. Bridge с указанным индексом должен быть заранее добавлен в конфигурацию.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-vpls)# bridge-group 25
```

description

Данной командой задаётся описание.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <LINE>
no description
```

Параметры

<LINE> – описание. Задается в виде строки длиной [1-255] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-PW

CONFIG-L2VPN-PW-CLASS

Пример

```
rtt(config-l2vpn-pw)# description PW_FOR_VLAN398
```

enable

Данной командой активируется выбранная функция.

Использование отрицательной формы команды (no) деактивирует выбранную функцию.

Синтаксис

```
[no] enable
```

Параметры

Команда не имеет параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-PW

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-pw)# enable
```

encapsulation mpls mtu

Данной командой устанавливается значение MTU для pseudowire-интерфейсов входящих в данный pw-class.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

```
encapsulation mpls mtu <MTU>
no encapsulation mpls mtu
```

Параметры

<MTU> – значение MTU, принимает значение в диапазоне [552..10000].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-PW-CLASS

Пример

```
rtt(config-l2vpn-pw-class)# encapsulation mpls mtu 1464
```

encapsulation mpls status-tlv disable

Данной командой отключается обмен status-tlv сообщениями.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] encapsulation mpls status-tlv disable
```

Параметры

Команда не имеет параметров.

Значение по умолчанию

status-tlv enable

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-PW-CLASS

Пример

```
rtt(config-l2vpn-pw-class)# encapsulation mpls status-tlv disable
```

ignore encapsulation-mismatch

Данная команда разрешает установку псевдопроводного соединения при несовпадении типа инкапсуляции.

Использование отрицательной формы команды (no) включает проверку типа инкапсуляции перед установкой псевдопроводного соединения.

Синтаксис

```
[ no ] ignore encapsulation-mismatch
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# ignore encapsulation-mismatch
```

ignore mtu-mismatch

Данная команда разрешает установку псевдопроводного соединения при несовпадении значения MTU.

Использование отрицательной формы команды (no) включает проверку значения MTU перед псевдопроводного соединения.

Синтаксис

```
[no] ignore mtu-mismatch
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# ignore mtu-mismatch
```

interface

С помощью данной команды задается Attached Circuit интерфейс.

Использование отрицательной формы команды (no) удаляет Attached Circuit интерфейс.

Синтаксис

```
interface { <IF> | <TUN> }
no interface
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-p2p)# interface gigabitethernet 1/0/4.398
```

mtu

Данная команда задает значение MTU для псевдопроводного соединения. Заданное значение передается в extended community.

Использование отрицательной формы команды (no) устанавливает значение MTU по умолчанию.

Синтаксис

```
mtu <VALUE>
no mtu
```

Параметры

<VALUE> – размер MTU, принимает значения в диапазоне [552..10000].

Значение по умолчанию

1500

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# mtu 1510
```

neighbor-address

С помощью данной команды указывается адрес маршрутизатора, до которого будет установлен псевдопровод.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
neighbor-address <ADDR>
no neighbor-address
```

Параметры

<ADDR> – IP-адрес маршрутизатора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Адрес соседа принимается равным LSR_ID.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-pw)# neighbor-address 192.168.24.78
```

p2p

С помощью данной команды создается L2VPN типа точка-точка (VPWS).

Использование отрицательной формы команды (no) удаляет указанный p2p.

Синтаксис

```
[no] p2p <NAME>
```

Параметры

<NAME> – имя p2p-сервиса, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN

Пример

```
rtt(config-l2vpn)# p2p VLAN_398
```

pw

С помощью данной команды создается псевдопровод.

Использование отрицательной формы команды (no) удаляет псевдопровод.

Синтаксис

```
pw <PW_ID> <LSR_ID>
no pw
```

Параметры

<PW_ID> – идентификатор psewdowire, задается в виде числа в диапазоне [1..4294967295];

<LSR_ID> – идентификатор LSR, до которого строится pseudo-wire, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-p2p)# pw 398 1.1.1.1
```

pw-class

С помощью данной команды создается pw-class.

Использование отрицательной формы команды (no) удаляет pw-class из конфигурации.

Синтаксис

```
[no] pw-class <WORD>
```

Параметры

<WORD> – имя pw-class длиной [1..31] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN

Пример

```
rtt(config-l2vpn)# pw-class Class1
```

pw-class

С помощью данной команды задается pw class для псевдопровода.

Использование отрицательной формы команды (no) убирает pw-class для данного псевдопровода.

Синтаксис

```
pw-class <WORD>
no pw-class
```

Параметры

<WORD> – имя pw-class, задается строкой длиной [1..31] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-PW

Пример

```
rtt(config-l2vpn-pw)# pw-class Class1
```

rd

Установка значения Route Distinguisher для данного экземпляра VPLS.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

Синтаксис

```
rd <RD>
no rd
```

Параметры

<RD> – значение Route distinguisher, задается в одном из следующем виде:

- <ASN>:<nn>, где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn>, где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4BASN>:<nn>, где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# rd 65000:10001
```

route-target

Установка перечня значений route-target extended community.

Синтаксис

```
[no] route-target { import | export | both } <RT>
```

Параметры

import – команда определяет маршруты из таблицы L2VPN со значением <RT>, которые будут помещены в соответствующий экземпляр VPLS;

export – команда устанавливает значения <RT>, с которыми маршруты из данного экземпляра VPLS будут анонсированы в BGP;

both – команда устанавливает значение <RT>, с которым маршруты будут анонсированы и импортированы в данном экземпляре VPLS;

<RT> – значение route-target, задается в одном из следующем виде:

- <ASN>:<nn>, где <ASN> – принимает значение [1..65535], nn – принимает значение [1..65535];
- <ADDR>:<nn>, где <ADDR> имеет вид – AAA.BBB.CCC.DDD/EE и AAA-DDD принимают значения [0..255], а nn – принимает значение [1..65535];
- <4BASN>:<nn>, где <4ASN> – принимает значение [1..4294967295], nn – принимает значение [1..65535].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# route-target export 65000:10001
```

show mpls l2vpn bindings

Данная команда позволяет вывести информацию по параметрам псевдопровода.

Синтаксис

```
show mpls l2vpn binding [ { [ local <LABEL-ID> ] [ remote <LABEL ID> ] | neighbor
<ADDR> | pw <PW-ID> | ve-id <VE-ID> } ]
```

Параметры

<LABEL-ID> – MPLS-метка, задается в виде числа в диапазоне [0..104875];

<ADDR> – IP-адрес LDP-соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PW-ID> – идентификатор psewdowire, задается в виде числа в диапазоне [1..4294967295];

<VE-ID> – идентификатор PE-маршрутизатора в домене VPLS [1.. 16384].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show mpls l2vpn bindings pw 100
Neighbor: 1.1.1.1, PW ID: 100
Local label: 19
Type: Eth Tagged
Group ID: 0
MTU: 1500
Remote label: 28
Type: Eth Tagged
Group ID: 0
MTU: 1500
```

show mpls l2vpn pseudowire

Данная команда позволяет вывести информацию по статусу псевдопровода.

Синтаксис

```
show mpls l2vpn pseudowire [ pw <PW-ID> ] [neighbor <ADDR> ]
```

Параметры

<PW-ID> – идентификатор psewdowire, задается в виде числа в диапазоне [1..4294967295];

<ADDR> – IP-адрес targeted LDP соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show mpls l2vpn pseudowire
Neighbor                        PW ID                    Type                Status
--------                      ----------              ----------            --------
1.1.1.1                          100                 Eth Tagged               Up
```

show mpls l2vpn p2p

Данная команда позволяет получить оперативную информацию по сервису VPWS.

Синтаксис

```
show mpls l2vpn p2p  [ name <WORD> ]
```

Параметры

<WORD> – имя сервиса, задается строкой длиной [1..31] символов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh mpls l2vpn p2p
P2P: VPWS
    gigabitethernet 1/0/4.10:
        MTU:    1500
        Status: Up
    PW ID 100, Neighbor 10.10.0.2:
        MTU:         1500
        Status TLV:  Enable
        Last change (d,h:m:s): 00,00:05:35
        Status:      Up
```

show mpls l2vpn pw-class

Данная команда позволяет получить оперативную информацию по шаблону конфигурации, примененному к псевдопроводу.

Синтаксис

```
show mpls l2vpn pw-class  [ name <WORD> ]
```

Параметры

<WORD> – имя шаблона, задается строкой длиной [1..31] символов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
PE1# sh mpls l2vpn pw-class
PW-class                        Neighbor   PW ID      Status Status-tlv MTU
------------------------------- ---------- --------- ------ ---------- ------
PW_template                     10.10.0.2  100        UP   Enable     1500
```

show mpls l2vpn vpls

Данная команда позволяет получить оперативную информацию по сервису VPLS.

Синтаксис

```
show mpls l2vpn vpls [ name <WORD> ]
```

Параметры

<WORD> – имя сервиса, задается строкой длиной [1..31] символов.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh mpls l2vpn vpls
VPLS: VPLS
    bridge 100:
        MTU:    1500
        Status: UP
    PWs:
      PW ID 23, Neighbor 10.10.0.1:
        MTU:         1500
        Status TLV:  Enable
        Last change(d,h:m:s): 00,19:00:07
        Status:      UP
```

transport-mode

Данная команда позволяет выбрать транспортный режим для данного псевдопровода.

Использование отрицательной формы команды (no) возвращает значение по умолчанию.

Синтаксис

```
transport-mode { ethenet | vlan }
no transport-mode
```

Параметры

ethernet – режим, при котором при входе в pseudo-wire из заголовка удаляется 802.1Q тег;

vlan – режим, при котором 802.1Q тег может быть сохранен при передаче через pseudo-wire.

Значение по умолчанию

ethernet

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-EOMPLS

Пример

```
rtt(config-l2vpn-p2p)# transport-mode vlan
```

ve id

Данной командой задается идентификатор экземпляра VPLS.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

Синтаксис

```
ve id <ID>
no ve id
```

Параметры

<ID> – идентификатор экземпляра VPLS, принимает значения в диапазоне [1..16384].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# ve id 5
```

ve range

Данной командой настраивается диапазон идентификаторов пограничных устройств VPLS.

Использование отрицательной формы команды (no) устанавливает значение ve range по умолчанию.

Синтаксис

```
ve range <ID
no ve range
```

Параметры

<ID> – идентификатор пограничного устройства, принимает значения в диапазоне [8..100].

Необходимый уровень привилегий

10

Значение по умолчанию

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# ve range 20
```

vpls

С помощью данной команды создается L2VPN типа point-to-multipoint.

Использование отрицательной формы команды (no) удаляет указанный vpls.

Синтаксис

```
[no] vpls <NAME>
```

Параметры

<NAME> – имя vpls-сервиса, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN

Пример

```
(config-l2vpn)# vpls Customer1
```

vpn id

Данной командой задается идентификатор VPN-соединения.

Использование отрицательной формы команды (no) удаляет параметр из конфигурации.

Синтаксис

```
vpn id <ID
no vpn id
```

Параметры

<ID> – идентификатор VPN-соединения, принимает значения в диапазоне [1..4294967295].

Необходимый уровень привилегий

10

Командный режим

CONFIG-L2VPN-AUTODISCOVERY-BGP

Пример

```
rtt(config-l2vpn-autodiscovery-bgp)# vpn id 5
```

23. РЕЗЕРВИРОВАНИЕ

23.1. Управление VRRP

clear vrrp-state

Данная команда останавливает выполнение протокола VRRP на устройстве, находящемся в состоянии Master. Это позволяет маршрутизатору в состоянии Backup инициировать процесс перехода в состояние Master и взять на себя управление виртуальным маршрутизатором.

Синтаксис

```
clear vrrp-state [ group { <GRID> | <GRID> vrf <VRF> } | id { <VRID> | <VRID> vrf
<VRF> } | vrf <VRF> ]
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32];
<VRID> – идентификатор VRRP-маршрутизатора, принимает значения [1..255];
<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# clear vrrp-state
```

Пример 2

```
rtt# clear vrrp-state group 32 vrf MGMT
```

show vrrp

Данная команда выводит информации о протоколе VRRP.

Синтаксис

```
show vrrp [ <ID> ] [ vrf <VRF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<ID> – номер процесса, принимает значения [1..255];
<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];
local – ключ для отображения информации о протоколе VRRP в формате вывода без включенного кластера.
При включенном кластере отображается информация для всех юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show vrrp
Virtual router   Virtual IP        Priority   Preemption   State    Inherit   Sync group ID
--------------   ---------------   --------   ----------   ------   -------   ------
1                172.16.0.1/32       100        Enabled      Master   --        --
2                172.17.0.1/32       100        Enabled      Master   --        --
```

Пример 2

```
rtt# show vrrp 1
VRRP ID:                        1
VRRP Inherit VRRP ID:           --
VRRP Version:                   2
Interface:                      gi1/0/3
Priority:                       100
State:                          Master
Track ID:                       --
Source IP address:              192.168.32.1
Virtual IP address:             172.16.0.1/32
Virtual MAC address:            00:00:5e:00:01:01
Synchronization group ID:       --
Preemption:                     Enabled, preemption delay is 0 seconds
Advertisement interval:         1 seconds
Gratuitous ARP delay:           5 seconds
Gratuitous ARP repeat:          5 times
Gratuitous ARP refresh:         0 seconds
Gratuitous ARP refresh repeat: 1 times
```

show vrrp inherit

Данная команда выводит информации о наследовании VRRP.

Синтаксис

```
show vrrp inherit [ vrf <VRF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения информации о наследовании VRRP в формате вывода без включенного кластера.

При включенном кластере отображается информация для всех юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show vrrp inherit
Virtual router   Inherit
--------------   ------------------------------------------------------------
2                1
```

vrrp

Данная команда создает VRRP-процесс на IP-интерфейсе.

Использование отрицательной формы команды (no) удаляет VRRP-процесс.

Синтаксис

```
[no] vrrp <NUMBER>
```

Параметры

<NUMBER> – номер VRRP-процесса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# vrrp 1
```

enable

Данной командой включаются VRRP-процесс на IP-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс на IP-интерфейсе.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

Проверка цели:

```
rtt(config-vrrp)# enable
```

authentication algorithm

Данная команда определяет алгоритм аутентификации.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации:

- cleartext – пароль, передается открытым текстом;
- md5 – пароль, хешируется по алгоритму md5.

Необходимый уровень привилегий

15

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# authentication algorithm cleartext
```

authentication key

Данная команда устанавливает пароль для аутентификации с соседом.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key
```

Параметры

<CLEAR-TEXT> – ключ, задаётся строкой от 1 до 8 символов;

<ENCRYPTED-TEXT> – зашифрованный ключ размером от 1 до 8 байт (от 2 до 16 символов). Задается в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# authentication key ascii-text 12345678
rtt(config-vrrp)# authentication key ascii-text encrypted CDE650
```

force-up

Данной командой устанавливается режим, когда виртуальный IP-интерфейс остается в состоянии UP вне зависимости от состояния самого интерфейса.

Использование отрицательной формы команды (no) устанавливает режим по умолчанию.

Синтаксис

```
[no] force-up
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# force-up
```

group

Данной командой устанавливается принадлежность VRRP-процесса к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов, так если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

Синтаксис

```
group <GRID>
no group
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# group 1
```

inherit-vrrp-id

Данной командой устанавливается наследование состояний от другого VRRP-процесса. Статус интерфейса наследника будет следовать состояниям VRRP родителя, идентификатор которого был задан.

Использование отрицательной формы команды (no) удаляет наследование VRRP.

Синтаксис

```
inherit-vrrp-id <VRID>
no inherit-vrrp-id
```

Параметры

<VRID> – идентификатор VRRP-маршрутизатора, принимает значения [1..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# inherit-vrrp-id 1
```

ip address

Данной командой устанавливается виртуальный IP-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IP-адрес маршрутизатора.

Синтаксис

```
ip address <ADDR/LEN > [ secondary ]
no ip address
```

Параметры

<ADDR/LEN> – виртуальный IP-адрес и длина маски, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32]. Можно указать несколько IP-адресов перечислением через запятую. Может быть назначено до 8 IP-адресов на интерфейс.
secondary – ключ для установки дополнительного IP-адреса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# ip address 172.16.0.1/32
rtt(config-vrrp)# ip address 172.16.0.2/32 secondary
```

preempt delay

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

Синтаксис

```
preempt delay <TIME>
no preempt delay
```

Параметры

<TIME> – время ожидания, определяется в секундах [0..1000].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# preempt delay 1
```

preempt disable

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IP-адрес совпадает с IP-адресом на интерфейсе, он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Также исключением является равенство значений vrrp priority, маршрутизатор не будет пытаться перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

Синтаксис

```
[no] preempt disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Переключение включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# preempt disable
```

priority

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

Синтаксис

```
priority <PR> [ unit <ID> ]
no priority [ unit <ID> ]
```

Параметры

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254].
<ID> – номер юнита для которого необходимо применить приоритет. Номер юнита принимает значения [1..4];

Значение по умолчанию

100

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# priority 1
rtt(config-vrrp)# priority 1 unit 1
rtt(config-vrrp)# priority 254 unit 2
```

source-ip

Данной командой устанавливается IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

Синтаксис

```
source-ip <IP>
no source-ip
```

Параметры

<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# source-ip 172.16.0.1
```

timers advertise

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers advertise <TIME>
no timers advertise
```

Параметры

<TIME> – время в секундах, принимает значения [1..40].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# timers advertise 2
```

timers garp delay

Данная команда определяет интервал, по истечении которого происходит отправка Gratuituous ARP сообщения(ий) при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers garp delay <TIME>
no timers garp delay
```

Параметры

<TIME> – время в секундах, принимает значения [1..60].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# timers garp delay 1
```

timers garp refresh

Данная команда определяет интервал, по истечении которого будет происходить периодическая отправка Gratuituous ARP-сообщения(ий), пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers garp refresh <TIME>
no timers garp refresh
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535].

Значение по умолчанию

Периодическая отправка отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# timers garp refresh 1
```

timers garp refresh-repeat

Данная команда определяет количество Gratuituous ARP-сообщений, которые будут отправляться с периодом garp refresh, пока маршрутизатор находится в состоянии Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers garp refresh-repeat <COUNT>
no timers garp refresh-repeat
```

Параметры

<COUNT> – количество сообщений, принимает значения [1..60].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# timers garp refresh-repeat 2
```

timers garp repeat

Данная команда определяет количество Gratuituous ARP сообщений, которые будут отправлены при переходе маршрутизатора в состояние Master.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers garp repeat <COUNT>
no timers garp repeat
```

Параметры

<COUNT> – количество сообщений, принимает значения [1..60].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# timers garp repeat 1
```

track-ip

Данная команда активирует управление статусом vrrp на основе icmp-reply от определенного IP-адреса.

Использование отрицательной формы команды (no) отключает управление статусом vrrp на основе icmp-reply от определенного IP-адреса.

Синтаксис

```
track-ip <ADDR>
no track-ip
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# track-ip 192.168.32.1
```

track-ip interval

Данная команда устанавливает интервал проверки при помощи icmp-request для управления статусом vrrp на основе icmp-reply от определенного IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение интервала по умолчанию.

Синтаксис

```
track-ip interval <TIME>
no track-ip interval
```

Параметры

<TIME> – интервал отправки icmp-request в секундах [3..60].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# track-ip interval 3
```

track-ip packets

Данная команда устанавливает количество icmp-request при каждой проверке для управления статусом vrrp на основе icmp-reply от определенного IP-адреса.

Использование отрицательной формы команды (no) устанавливает значение количества пакетов по умолчанию.

Синтаксис

```
track-ip packets <COUNT>
no track-ip packets
```

Параметры

<COUNT> – интервал отправки icmp-request в секундах [1..5].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# track-ip packets 1
```

version

Данной командой задаётся версия VRRP-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
version <VERSION>
no version
```

Параметры

<VERSION> – версия VRRP-протокола: 2, 3.

Значение по умолчанию

2

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# version 3
```

23.2. Управление IPv6 VRRP

ipv6 vrrp

Данная команда создает VRRP-процесс на IPv6-интерфейсе.

Использование отрицательной формы команды (no) удаляет VRRP-процесс.

Синтаксис

```
[no] ipv6 vrrp <NUMBER>
```

Параметры

<NUMBER> – номер VRRP-процесса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 vrrp 1
```

enable

Данной командой включаются VRRP-процесс на IP-интерфейсе.

Использование отрицательной формы команды (no) выключает VRRP-процесс на IP-интерфейсе.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

Проверка цели:

```
rtt(config-ipv6-vrrp)# enable
```

group

Данной командой устанавливается принадлежность VRRP-процесса к группе. Группа предоставляет возможность синхронизировать несколько VRRP-процессов, так, если в одном из процессов произойдет смена мастера, то в другом процессе также произведется смена ролей.

Использование отрицательной формы команды (no) удаляет VRRP-маршрутизатор из группы.

Синтаксис

```
group <GRID>
no group
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# group 1
```

inherit-vrrp-id

Данной командой устанавливается наследование состояний от другого VRRP-процесса. Статус интерфейса наследника будет следовать состояниям VRRP родителя, идентификатор которого был задан.

Использование отрицательной формы команды (no) удаляет наследование VRRP.

Синтаксис

```
inherit-vrrp-id <VRID>
no inherit-vrrp-id
```

Параметры

<VRID> – идентификатор VRRP-маршрутизатора, принимает значения [1..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# inherit-vrrp-id 1
```

ipv6 address

Данной командой устанавливается виртуальный IPv6-адрес VRRP-маршрутизатора.

Использование отрицательной формы команды (no) удаляет виртуальный IPv6-адрес маршрутизатора.

Синтаксис

```
ipv6 address <IPV6-ADDR> [secondary]
no ipv6 address
```

Параметры

<IPV6-ADDR> – виртуальный IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать несколько IPv6-адресов перечислением через запятую. Может быть назначено до 8 IPv6-адресов на интерфейс;
secondary – ключ для установки дополнительного IP-адреса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# ipv6 address fe80::abcd:1234:5678:9abc/64
```

preempt delay

Данной командой задается временной интервал, по истечении которого Backup-маршрутизатор с более высоким приоритетом будет пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

Синтаксис

```
preempt delay <TIME>
no preempt delay
```

Параметры

<TIME> – время ожидания, определяется в секундах [0..1000].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# preempt delay 1
```

preempt disable

Данной командой определяется, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.

Исключением является маршрутизатор, у которого виртуальный IP-адрес совпадает с IP-адресом на интерфейсе, он всегда будет перехватывать на себя роль Master вне зависимости от данной настройки.

Также исключением является равенство значений vrrp priority, маршрутизатор не будет пытаться перехватывать на себя роль Master вне зависимости от данной настройки.

Использование отрицательной формы команды (no) восстанавливает настройки по умолчанию.

Синтаксис

```
[no] preempt disable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Переключение включено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# preempt disable
```

priority

Данной командой устанавливается приоритет VRRP-маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение приоритета по умолчанию.

Синтаксис

```
priority <PR> [ unit <ID> ]
no priority [ unit <ID> ]
```

Параметры

<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254];

<ID> – номер юнита, для которого необходимо применить приоритет. Номер юнита принимает значения [1..4].

Значение по умолчанию

100

Необходимый уровень привилегий

10

Командный режим

CONFIG-VRRP

Пример

```
rtt(config-vrrp)# priority 1
rtt(config-vrrp)# priority 1 unit 1
rtt(config-vrrp)# priority 254 unit 2
```

source-ip

Данной командой устанавливается IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

Синтаксис

```
source-ip <IP>
no source-ip
```

Параметры

<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# source-ip fc00::1
```

timers advertise

Данная команда определяет интервал между отправкой VRRP-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers advertise <TIME>
no timers advertise
```

Параметры

<TIME> – время в секундах, принимает значения [1..40].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# timers advertise 2
```

timers nd delay

Данная команда определяет задержку между установлением IPv6 VRRP состояния MASTER и началом рассылки ND-сообщений.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers nd delay <TIME>
no timers nd delay
```

Параметры

<TIME> – время в секундах, принимает значения [1..60].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# timers nd delay 1
```

timers nd refresh

Данная команда определяет период обновления информации протокола ND для IPv6 VRRP в состоянии MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers nd refresh <TIME>
no timers nd refresh
```

Параметры

<TIME> – время в секундах, принимает значения [1..65535].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# timers nd refresh 1
```

timers nd refresh-repeat

Данная команда определяет количество ND-сообщений, отправляемых за период обновления для IPv6 VRRP в состоянии MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers nd refresh-repeat <NUM>
no timers nd refresh-repeat
```

Параметры

<NUM> – количество, принимает значения [1..60].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# timers nd refresh-repeat 1
```

timers nd repeat

Данная команда определяет количество отправок ND-пакетов после установки IPv6 VRRP в состояние MASTER.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
timers nd repeat <NUM>
no timers nd repeat
```

Параметры

<NUM> – количество, принимает значения [1..60].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-VRRP

Пример

```
rtt(config-ipv6-vrrp)# timers nd repeat 2
```

show ipv6 vrrp

Данная команда выводит информации о протоколе VRRP.

Синтаксис

```
show ipv6 vrrp [ <ID> ] [ vrf <VRF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<ID> – номер процесса, принимает значения [1..255];
<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];
local – ключ для отображения информации о протоколе VRRP в формате вывода без включенного кластера.
При включенном кластере отображается информация для всех юнитов кластера.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show ipv6 vrrp
 Virtual router   Virtual IP                                  Priority   Preemption   State    Inherit   Sync group ID
 --------------   ------------------------------------------  --------   ----------   ------   -------   -------------
 1                fe80::1/112, fc00::254/112                  100        Enabled      Master   --        --
```

Пример 2

```
rtt# show ipv6 vrrp 1
VRRP ID:                        1
VRRP Inherit VRRP ID:           --
VRRP Version:                   3
Interface:                      gi2/0/4
Priority:                       100
State:                          Master
Track ID:                       --
Source IP address:              fe80::aaf9:4bff:feac:e8a0
Virtual IP addresses:           fe80::1/112
                                fc00::254/112
Virtual MAC address:            00:00:5e:00:01:02
Synchronization group ID:       --
Preemption:                     Enabled, preemption delay is 0 seconds
Advertisement interval:         1 seconds
Gratuitous ARP delay:           5 seconds
Gratuitous ARP repeat:          5 times
Gratuitous ARP refresh:         0 seconds
Gratuitous ARP refresh repeat: 1 times
```

23.3. Настройка MultiWAN

description

Данной командой определяется описание правила.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание правила WAN, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

wan load-balance target-listCONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

Пример

```
rtt(config-wan-rule)# description "tunnel to branch"
```

enable

Данной командой включаются правило WAN, проверка цели.

Использование отрицательной формы команды (no) выключает правило WAN, проверку цели.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-RULE

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-RULE

CONFIG-IPV6-WAN-TARGET

Пример

Проверка цели:

```
rtt(config-ipv6-wan-rule)# enable
```

extended count

Данной командой определяется количество одновременно отправленных ICMP-запросов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended count <NUM>
no extended count
```

Параметры

<NUM> – количество одновременно отправленных ICMP-запросов, принимает значение в диапазоне [10..100].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended count 75
```

extended jitter

Данная команда определяет пороговое значение джиттера, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended jitter <TIME>
no extended jitter
```

Параметры

<TIME> – пороговое значение джиттера в миллисекундах, принимает значение в диапазоне [50-1000].

Значение по умолчанию

50

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended jitter 78500
```

extended loss

Данная команда определяет пороговое значение процента потерь, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended loss <NUM>
no extended loss
```

Параметры

<NUM> – пороговое значение процента потерь [1..100].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended loss 80
```

extended period

Данная команда определяет период отправки ICMP-запросов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended period <TIME>
no extended period
```

Параметры

<TIME> – период отправки ICMP запросов в секундах, принимает значение в диапазоне [10..600].

Значение по умолчанию

240

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended period 120
```

extended rtt

Данная команда определяет максимальное пороговое значение круговой задержки, при превышении которого качество канала является неудовлетворительным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended rtt <TIME>
no extended rtt
```

Параметры

<TIME> – максимальное пороговое значение RTT в миллисекундах, принимает значение в диапазоне [200..1000].

Значение по умолчанию

240

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended rtt 800
```

extended timeout

Данная команда определяет максимальное время ожидания ответа на ICMP-запрос.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
extended timeout <TIME>
no extended timeout
```

Параметры

<TIME> – максимальное время ожидания ответа на ICMP-запрос в миллисекундах, принимает значение в диапазоне [500..10000].

Значение по умолчанию

1000

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# extended timeout 5000
```

failover

Данной командой осуществляется переключение из режима балансировки в режим резервирования.

Использование отрицательной формы команды (no) возвращает режим балансировки.

Синтаксис

```
[no] failover
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

Пример

```
rtt(config-wan-rule)# failover
```

ip address

Данной командой указывается IP-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

Синтаксис

```
ip address <ADDR>
no ip address
```

Параметры

<ADDR> – IP-адрес назначения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# ip address 18.168.1.25
```

ipv6 address

Данной командой указывается IPv6-адрес проверки.

Использование отрицательной формы команды (no) удаляет указанный адрес.

Синтаксис

```
ipv6 address <IPV6-ADDR>
no ipv6 address
```

Параметры

<IPV6-ADDR> – IPv6-адрес назначения, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-WAN-TARGET

Пример

```
rtt(config-ipv6-wan-target)# ipv6 address fc00::2
```

ipv6 wan load-balance enable

Данной командой включается WAN режим на интерфейсе для IPv6-стека.

Использование отрицательной формы команды (no) выключает WAN-режим для IPv6-стека.

Синтаксис

```
[no] ipv6 wan load-balance enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 wan load-balance enable
```

ipv6 wan load-balance failure-count

Данной командой определяется количество неудачных попыток проверки соединения через IPv6-стек, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 wan load-balance failure-count <VALUE>
no ipv6 wan load-balance failure-count
```

Параметры

<VALUE> – количество попыток, определяется в диапазоне [1…10].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 wan load-balance failure-count 3
```

ipv6 wan load-balance nexthop

Данной командой определяется IPv6-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес соседа.

Синтаксис

```
ipv6 wan load-balance nexthop <IPV6-ADDR>
no ipv6 wan load-balance nexthop
```

Параметры

<IPV6-ADDR> – IPv6-адрес назначения (шлюз), задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config-if-gi)#ipv6 wan load-balance nexthop 220::71
```

ipv6 wan load-balance rule

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила для протокола IPv6.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

Синтаксис

```
ipv6 wan load-balance rule <ID>
no ipv6 wan load-balance rule { <ID> | all }
```

Параметры

<ID> – идентификатор создаваемого правила, принимает значения: [1..50].

Значение «all» используется при удалении всех правил WAN для протокола IPv6.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 wan load-balance rule 1
```

ipv6 wan load-balance source-address

Данной командой определяется IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых ICMPv6-пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес источника.

Синтаксис

```
ipv6 wan load-balance source-address <IPV6-ADDR>
no ipv6 wan load-balance source-address
```

Параметры

<IPV6-ADDR> – IPv6-адрес источника, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LT

Пример

```
rtt(config-if-gi)#ipv6 wan load-balance source-address 220::71
```

ipv6 wan load-balance success-count

Данной командой определяется количество успешных попыток проверки соединения по протоколу IPv6, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 wan load-balance success-count <VALUE>
no ipv6 wan load-balance success-count
```

Параметры

<VALUE> – количество попыток, определяется в диапазоне [1…10].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config-if-gi)# ipv6 wan load-balance success-count 3
```

ipv6 wan load-balance target-list

Данной командой создается список IPv6-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка.

Использование отрицательной формы команды (no) удаляет созданный список.

Синтаксис

```
ipv6 wan load-balance target-list <NAME>
no ipv6 wan load-balance target-list { <NAME> | all }
```

Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IPv6-адресов для проверки целостности соединения.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 wan load-balance target-list ten1
```

ipv6 wan load-balance target-list

Данной командой производится привязка списка IPv6-адресов для проверки целостности соединения на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет список с интерфейса.

Синтаксис

```
ipv6 wan load-balance target-list <NAME>
no ipv6 wan load-balance target-list (удаление привязки списка в режиме
конфигурирования сетевых интерфейсов)
```

Параметры

<NAME> – название списка, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config)# ipv6 wan load-balance target-list ten1
```

ipv6 wan load-balance target-list check-all

Данной командой будут проверяться все IPv6-адреса из списка проверки целостности.

Использование отрицательной формы команды (no) отменяет проверку всех IPv6-адресов из списка проверки целостности. В случае недоступности одного из проверяемых узлов, шлюз будет считаться недоступным.

Синтаксис

```
[no] ipv6 wan load-balance target-list check-all
```

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LT

Пример

```
rtt(config-if-te)# ipv6 wan load-balance target-list check-all
```

mode

Данная команда определяет режим проверки доступности цели для проверки состояния канала.

В рамках одного целевого листа (wan load-balance target-list) допустимо использовать только цели (target) с одинаковым режимом работы (mode basic или mode extended).

В рамках одного процесса MultiWAN при использовании 2 или более wan load-balance target-list допустимо использовать цели (target) только с одинаковым режимом работы (mode basic или mode extended).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode { basic | extended }
no mode
```

Параметры

basic – базовый, регулируется допустимым значением положительных ответов на ICMP-запросы;

extended – расширенный метод, в нем производится оценка уровня потерь, задержки, джиттер.

Значение по умолчанию

basic

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

Пример

```
rtt(config-wan-target)# mode extended
```

outbound

Данной командой определяются интерфейсы или туннели, которые являются шлюзами в маршруте, создаваемом службой MultiWAN. Количество шлюзов в маршруте зависит от режима работы MultiWAN:

- при балансировке в список nexthop-маршрута добавляются IP-адреса шлюзов (раздел wan load-balance nexthop) всех активных интерфейсов;
- при резервировании в качестве nexthop-маршрута выбирается IP-адрес шлюза (раздел wan load-balance nexthop) активного интерфейса с наибольшим весом.

Использование отрицательной формы команды (no) исключает указанный интерфейс или туннель из правила MultiWAN.

Синтаксис

```
[no] outbound { interface <IF> | tunnel <TUN> } [<WEIGHT> ]
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора (только для IPv4);

<WEIGHT> – вес туннеля или интерфейса, определяется в диапазоне [1..255]. Если установить значение 2, то по данному интерфейсу будет передаваться в 2 раза больше трафика, чем по интерфейсу с дефолтным значением. В режиме резервирования активным будет маршрут с наибольшим весом.

Значение по умолчанию

WEIGHT – 1

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-RULE

CONFIG-IPV6-WAN-RULE

Пример 1

```
rtt(config-wan-rule)# outbound interface gigabitethernet 1/0/15
```

Пример 2

```
rtt(config-ipv6-wan-rule)# outbound interface bridge 2
```

resp-time

Данной командой определяется время ожидания ответа на запрос по протоколу ICMP.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
resp-time <TIME>
no resp-time
```

Параметры

<TIME> – время ожидания, определяется в секундах [1..30].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET

CONFIG-IPV6-WAN-TARGET

Пример

```
rtt(config-wan-target)# resp-time 3
```

show wan rules

Данная команда используется для просмотра оперативной информации по правилам WAN для протокола IPv4.

Синтаксис

```
show wan rules <ID>
```

Параметры

<ID> – номер правила WAN, принимает значения [1..50].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# sh wan rules 1
Rule 1 detailed information:
    Failover:     Enabled
    Network: 11.11.11.0/24 Metric: 50
        gi1/0/1 Weight: 2 Nexthop: 3.3.3.1 [Active]
```

show wan cellular status modem

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv4-стека.

Синтаксис

```
show wan cellular status modem [ <MODEM-ID> ]
```

Параметры

<MODEM-ID> – идентификатор ранее созданного 2G/3G/4G-модема (только для IPv4).

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы с включенным WAN-режимом для IPv4-стека. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для IPv4-стека.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show wan cellular status modem 1
Cellular 'modem 1' detailed information:
    VRF:                          default
    State:                        Active
    Nexthop address:              192.168.8.1 (dhcp)
    Last time success (d,h:m:s): 00,00:00:00
    Last time failure (d,h:m:s): 00,00:02:24
    Uptime (d,h:m:s):             00,00:02:23
    Targets:
        8.8.4.4 Test/Ping State: pass Attempts: 3/3
        8.8.4.4 Test/Ping State: -- Attempts: 0/5
```

show wan interfaces status

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv4-стека.

Синтаксис

```
show wan interfaces status [ <IF> ]
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы с включенным WAN-режимом для IPv4-стека. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для IPv4-стека.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show wan  interfaces status gigabitethernet 1/0/1
Interface 'gigabitethernet 1/0/1' detailed information:
    State:             Active
    Nexthop address:   3.3.3.1
    Last time success (d,h:m:s): 00,00:00:00
    Last time failure (d,h:m:s): 00,01:48:09
    Uptime (d,h:m:s):            00,01:48:08
    Targets:
        3.3.3.1 Test/Ping State: pass Attempts: 1/1
```

show wan tunnels status

Данная команда используется для просмотра оперативной информации о состоянии туннелей с включенным WAN-режимом для IPv4-стека.

Синтаксис

```
show wan tunnels status [ <TUN> ]
```

Параметры

<INDEX> – идентификатор туннеля, задается в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

В команде можно указать несколько туннелей. Если не указывать индексы туннелей, то будут отображены статусы заданной группы всех туннелей с включенным WAN-режимом для IPv4-стека. Если задан конкретный туннель, то будет отображена детальная информация по данному туннелю. При выполнении команды без параметра будут показаны статусы всех туннелей с включенным WAN-режимом для IPv4-стека.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show wan tunnels status gre 1
Tunnel 'gre 1' detailed information:
    State:             Active
    Nexthop address:   5.5.5.2
    Last time success (d,h:m:s): 00,00:00:00
    Last time failure (d,h:m:s): 00,00:50:02
    Uptime (d,h:m:s):            00,00:50:01
    Targets:
        5.5.5.2 Test/Ping State: pass Attempts: 1/1
```

show ipv6 wan rules

Данная команда используется для просмотра оперативной информации по правилам WAN для протокола IPv6.

Синтаксис

```
show ipv6 wan rules <ID>
```

Параметры

<ID> – номер правила WAN, принимает значения [1..50].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 wan rules 1
Rule 1 detailed information:
    Failover:     Enabled
    Network: 2a14::/120 Metric: 0
        bridge 1 Weight: 1 Nexthop: 2a12::1 [Active]
```

show ipv6 wan interfaces status

Данная команда используется для просмотра оперативной информации о состоянии интерфейсов с включенным WAN-режимом для IPv6-стека.

Синтаксис

```
show ipv6 wan interfaces status [ <IF> ]
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

В команде можно указать несколько интерфейсов. Если не указывать индексы интерфейсов, то будут отображены статусы всех интерфейсов заданной группы интерфейсов с включенным WAN-режимом для стека IPv6. Если задан конкретный интерфейс, то будет отображена детальная информация по данному интерфейсу. При выполнении команды без параметра будут показаны статусы всех системных интерфейсов с включенным WAN-режимом для стека IPv6.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 wan interfaces status bridge 1
Interface 'bridge 1' detailed information:
    State:              Active
    Nexthop address:    2a12::1
    Source address:     fe80::aaf9:4bff:feaa:2dcf
    Last time success (d,h:m:s): 00,00:00:00
    Last time failure (d,h:m:s): 00,00:19:29
    Uptime (d,h:m:s):            00,00:19:28
    Targets:
        2a12::1 Test/Ping State: pass Attempts: 1/1
```

stickiness

Данной командой включается отправка ответных пакетов сессии через тот же интерфейс, через который получены входящие пакеты сессии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
stickiness
no stickiness
```

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-RULE

Пример

```
rtt(config-wan-rule)# stickiness
```

target

Данной командой создается цель проверки и осуществляется переход в режим настройки параметров цели.

Использование отрицательной формы команды (no) удаляет созданную цель.

Синтаксис

```
target <ID>
no target { <ID> | all }
```

Параметры

<ID> – идентификатор цели, задаётся в пределах [1..50]. Если при удалении используется значение параметра «all», то будут удалены все цели для конфигурируемого списка целей.

all – удалить все цели.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WAN-TARGET-LIST

CONFIG-IPV6-WAN-TARGET-LIST

Пример

```
rtt(config-target-list)# target 1
```

wan load-balance enable

Данной командой включается WAN-режим на интерфейсе для IPv4-стека.

Использование отрицательной формы команды (no) выключает WAN-режим для IPv4-стека.

Синтаксис

```
[no] wan load-balance enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-if-gi)# wan load-balance enable
```

wan load-balance failure-count

Данной командой определяется количество неудачных попыток проверки соединения, после которых, при отсутствии ответа от встречной стороны, соединение считается неактивным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
wan load-balance failure-count <VALUE>
no wan load-balance failure-count
```

Параметры

<VALUE> – количество попыток, определяется в диапазоне [1…10].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-if-gi)# wan load-balance failure-count 3
```

wan load-balance nexthop

Данной командой определяется IP-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес соседа.

Синтаксис

```
wan load-balance nexthop { <ADDR> | dhcp enable | tunnel enable }
no wan load-balance nexthop
```

Параметры

<ADDR> – IP-адрес назначения (шлюз), задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

dhcp enable – если на интерфейсе IP-адрес получен через DHCP-клиента, используется шлюз с DHCP-сервера.

tunnel enable – использовать в качестве nexthop – peer адрес. Применимо для подключаемых интерфейсов, работающих через PPP.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-gre)# wan load-balance nexthop 16.168.1.25
```

wan load-balance rule

Данной командой создается правило WAN и осуществляется переход в режим настройки параметров правила.

Использование отрицательной формы команды (no) удаляет созданное WAN-правило.

Синтаксис

```
wan load-balance rule <ID>
no wan load-balance rule { <ID> | all }
```

Параметры

<ID> – идентификатор создаваемого правила, принимает значения: [1..50].

Значение «all» используется при удалении всех правил WAN для протокола IPv6.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# wan load-balance rule 1
```

wan load-balance source-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых ICMP-пакетах для проверки целей.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

Синтаксис

```
wan load-balance source-address <ADDR>
no wan load-balance source-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LT

Пример

```
rtt(config)# wan load-balance source-address 16.168.1.25
```

wan load-balance success-count

Данной командой определяется количество успешных попыток проверки соединения, после которых, в случае успеха, соединение считается вновь активным.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
wan load-balance success-count <VALUE>
no wan load-balance success-count
```

Параметры

<VALUE> – количество попыток, определяется в диапазоне [1…10].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-if-gi)# wan load-balance success-count 3
```

wan load-balance target-list

Данной командой создается список IP-адресов для проверки целостности соединения и осуществляется переход в режим настройки параметров списка.

Использование отрицательной формы команды (no) удаляет созданный список.

Синтаксис

```
wan load-balance target-list <NAME>
no wan load-balance target-list { <NAME> | all }
```

Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IP-адресов для проверки целостности соединения.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# wan load-balance target-list ten1
```

wan load-balance target-list

Данной командой производится привязка списка IP-адресов для проверки целостности соединения на сетевом интерфейсе.

Использование отрицательной формы команды (no) удаляет созданный список.

Синтаксис

```
wan load-balance target-list <NAME>
no wan load-balance target-list
```

Параметры

<NAME> – название списка, задается строкой до 31 символа. Значение «all» используется при удалении всех списков IP-адресов для проверки целостности соединения.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config)# wan load-balance target-list ten1
```

wan load-balance target-list check-all

Данной командой будут проверяться все IP-адреса из списка проверки целостности. В случае недоступности одного из проверяемых узлов шлюз будет считаться недоступным.

Использование отрицательной формы команды (no) отменяет проверку всех IP-адресов из списка проверки целостности.

Синтаксис

```
[no] wan load-balance target-list check-all
```

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-LT

CONFIG-CELLULAR-MODEM

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-PPPOE

CONFIG-PPTP

Пример

```
rtt(config-gre)# wan load-balance target-list check-all
```

23.4. Настройка объектов отслеживания событий

delay down

Данной командой устанавливается задержка перед переходом объекта отслеживания в неактивное состояние.

Использование отрицательной формы команды (no) убирает задержку перед переходом объекта отслеживания в неактивное состояние.

Синтаксис

```
delay down <TIME>
[no] delay down
```

Параметры

<TIME> – время задержки в секундах перед переходом объекта отслеживания в неактивное состояние, принимает значение [1..300].

Значение по умолчанию

По умолчанию задержка отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# delay down 5
```

delay up

Данной командой устанавливается задержка перед переходом объекта отслеживания в активное состояние.

Использование отрицательной формы команды (no) убирает задержку перед переходом объекта отслеживания в активное состояние.

Синтаксис

```
delay up <TIME>
[no] delay up
```

Параметры

<TIME> – время задержки в секундах перед переходом объекта отслеживания в неактивное состояние, принимает значение [1..300].

Значение по умолчанию

По умолчанию задержка отключена.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# delay up 5
```

description

Данной командой определяется описание объекта отслеживания событий.
Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание объекта отслеживания событий, задаётся строкой до 255 символов.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# description "check virtual router state"
```

enable

Данной командой включается объект отслеживания.
Использование отрицательной формы команды (no) отключает объект отслеживания.

Синтаксис

```
[no] enable
```

Значение по умолчанию

Объект выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# enable
```

enable track

Данной командой активируется функционал по состоянию объекта отслеживания.
Использование отрицательной формы команды (no) отключает объект отслеживания.

Синтаксис

enable track <ID>

[no] enable

Значение по умолчанию

Объект выключен.

Необходимый уровень привилегий

10
Командный режим

CONFIG-IPSEC-VPN

Пример

```
rtt(config-ipsec-vpn)# enable track 10
```

mode

Данной командой задается режим работы объекта отслеживания.
Использование отрицательной формы команды (no) переводит режим на значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – условие нахождения объекта отслеживания в активном состоянии, принимает значения:

- and – объект будет находиться в активном состоянии, если все отслеживаемые условия будут выполняться;
- or – объект будет находиться в активном состоянии, если хотя бы одно отслеживаемое условие будет выполняться.

Значение по умолчанию

and

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# mode or
```

show tracks

Данной командой выводится информация о текущем статусе всех сконфигурированных объектов отслеживания.

Синтаксис

```
show tracks
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show tracks
Track ID   State   Last change     Description
                   (d,h:m:s)
--------   -----   -------------   ------------------------------------
1          Up      00,00:00:04     DNS probe state
2          Down    00,00:36:04     VRRP-1 Master
```

show track

Данной командой выводится информация об объекте отслеживания.

Синтаксис

```
show track <TRACK-ID>
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, принимает значение [1..100].

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show track 1
Track 1:
  State:         Up
  Changes count: 1 (last 00,00:00:07)
  Mode:          And
  Delay up:      0s
  Delay down:    0s
  Description:   DNS probe state

Conditions:

Type    ID      State   Mode             Last change     VRF
                                         (d,h:m:s)
----    -----   -----   --------------   -------------   ----------------
SLA     1       True    State success    00,00:00:07     --
```

shutdown track

Данной командой выполняется управление состоянием интерфейса в зависимости от состояния объекта отслеживания.
Использование отрицательной формы команды (no) отменяет управление состоянием интерфейса.

Синтаксис

```
shutdown track <ID>
no shutdown track
```

Параметры

<ID> – номер объекта отслеживания, принимает значения в диапазоне [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# shutdown track 2
```

track

Данная команда используется для создания объекта отслеживания (tracking-объекта) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный объект отслеживания.

Синтаксис

```
[no] track <TRACK-ID>
```

Параметры

<TRACK-ID> – идентификатор объекта отслеживания, принимает значение [1..100].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# track 10
```

track sla test

Данной командой устанавливается слежение за состоянием определенного sla-теста.
Использование отрицательной формы команды (no) отключает слежение за состоянием sla-теста.

Синтаксис

```
track sla test <ID> [mode <MODE> ]
[no] track sla test <ID>
```

Параметры

<ID> – идентификатор sla-теста, принимает значение в диапазоне [1..10000].
<MODE> – режим слежения за sla-тестом, может принимать значения:

- state success – отслеживается успешное состояние sla-теста;
- state fail – отслеживается провальное состояние sla-теста;
- reachability – отслеживаются успешность прохождения контрольной фазы с SLA-Responder/доступность удаленного узла при icmp-тестировании.

При указании команды без аргумента mode по умолчанию устанавливается значение mode state success.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# track sla test 1 mode state success
```

track vrrp id

Данной командой устанавливается слежение за текущим состоянием (ролью) определенного VRRP-маршрутизатора.
При использовании модификатора «not» правило будет срабатывать для всех состояний, за исключением указанного.

Использование отрицательной формы команды (no) отключает слежение за состоянием VRRP-маршрутизатора.

Синтаксис

```
track vrrp id <VRID> state [not] <STATE> { master | backup | fault } [ vrf <VRF> ]
no track vrrp id <VRID> [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

<VRID> – идентификатор отслеживаемого VRRP-маршрутизатора, принимает значения [1..255];

<STATE> – состояние VRRP-маршрутизатора, при котором объект отслеживания будет менять своё состояние. Может принимать значения:

- master – роль master;
- backup – роль backup;
- fault – fault-состояние VRRP-процесса;

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# track vrrp id 2 state master
```

track interface

Данной командой устанавливается слежение за административным состоянием определенного интерфейса.
Использование отрицательной формы команды (no) отключает слежение за состоянием интерфейса.

Синтаксис

```
track interface <IF> [ state <STATE> ]
[no] track interface <IF>
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;
<STATE> – режим слежения за интерфейсом, может принимать значения:

- up – административное состояние "Up";
- down – административное состояние "Down".

При указании команды без аргумента state по умолчанию устанавливается значение state up.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# track interface gigabitethernet 1/0/1 state up
```

track tunnel

Данной командой устанавливается слежение за административным состоянием определенного туннеля. Использование отрицательной формы команды (no) отключает слежение за состоянием туннеля.

Синтаксис

```
track tunnel <TUN> [ state <STATE> ]
[no] track tunnel <TUN>
```

Параметры

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.
<STATE> – режим слежения за туннелем, может принимать значения:

- up – административное состояние "Up";
- down – административное состояние "Down".

При указании команды без аргумента state по умолчанию устанавливается значение state up.

Необходимый уровень привилегий

10

Командный режим

CONFIG-TRACK

Пример

```
rtt(config-track)# track tunnel openvpn 1 state down
```

vrrp priority track

Данной командой добавляется условие изменения приоритета VRRP-процесса, вступающее в силу при активном состоянии объекта отслеживания.
Использование отрицательной формы команды (no) удаляет условие.

Синтаксис

```
vrrp priority track <ID> { <PRIO> | increment <INC> | decrement <DEC> }
no vrrp priority track
```

Параметры

<ID> – номер объекта отслеживания, принимает значения в диапазоне [1..100];
<PRIO> – значение приоритета, которое выставится VRRP-процессу при условии активного состояния

объекта отслеживания, принимает значения в диапазоне [1..254];
<INC> – значение, на которое увеличится приоритет VRRP-процесса при условии активного состояния объекта отслеживания, принимает значения в диапазоне [1..254];
<DEC> – значение, на которое уменьшится приоритет VRRP-процесса при условии активного состояния объекта отслеживания, принимает значения в диапазоне [1..254].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# vrrp priority track 2 increment 20
```

23.5. Управление Dual-Homing

В текущей версии ПО данный функционал поддерживается только на маршрутизаторах R800.

backup interface

Данной командой указывается резервный интерфейс, на который будет происходить переключение при потере связи на основном. Включение резервирования возможно только на тех интерфейсах, на которых отключен протокол Spanning Tree и включен VLAN Ingress Filtering.

Использование отрицательной формы команды (no) удаляет настройку с интерфейса.

Синтаксис

```
backup interface <IF> vlan <VID>
no backup interface
```

Параметры

<IF> – интерфейс, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094]. Можно также задать диапазоном через «-» или перечислением через «,».

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-TE

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# backup interface gigabitethernet 1/0/15 vlan 10-200
```

backup-interface mac-duplicate

Данной командой указывается количество копий пакетов с одним и тем же MAC-адресом, которые будут отправлены в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
backup-interface mac-duplicate <COUNT>
no backup-interface mac-duplicate
```

Параметры

<COUNT> – количество копий пакетов, принимает значение [1..4].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# backup-interface mac-duplicate 4
```

backup-interface mac-per-second

Данной командой указывается количество пакетов в секунду, которое будет отправлено в активный интерфейс при переключении.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
backup-interface mac-per-second <COUNT>
no backup-interface mac-per-second
```

Параметры

<COUNT> – количество пакетов в секунду, принимает значение [50..400].

Значение по умолчанию

400

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# backup-interface mac-per-second 200
```

backup-interface preemption

Данной командой указывается, что необходимо осуществить переключение на основной интерфейс при восстановлении связи. Если настроено восстановление основного интерфейса при активном резервном, то тогда при поднятии линка на основном интерфейсе трафик будет переключен на него.

Использование отрицательной формы команды (no) восстанавливает настройку по умолчанию.

Синтаксис

```
[no] backup-interface preemption
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Переключение отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# backup-interface preemption
```

show interfaces backup

Данная команды выводит информацию о состоянии основного и резервного интерфейса.

Синтаксис

```
show interfaces backup
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces backup
Backup Interface Options:
    Preemption isDisabled
    MAC recovery packets rate400 pps
    Recovery packets repeats count1
Master Interface         Backup Interface            State                      VID
----------------------   -------------------------   ----------------------    -----
gigabitethernet 1/0/8    gigabitethernet 1/0/7       Master Standby/Backup Up   10
```

```
----------------------   -------------------------   ----------------------    -----
gigabitethernet 1/0/7    gigabitethernet 1/0/8       Master Up/Backup Standby   20
----------------------   -------------------------   ----------------------    -----
```

23.6. Настройка failover-сервисов

23.6.1. Настройка общих параметров для failover-сервисов

ip failover

Данной командой осуществляется переход в режим конфигурирования общих параметров для failover-сервисов.

Использование отрицательной формы команды (no) удаляет настройки общих параметров для failover-сервисов.

Синтаксис

```
[no] ip failover [ vrf <VRF> ]
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа.

Значение по умолчанию

Отключено

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip failover
rtt(config-failover)#
```

local-address

Данной командой устанавливается IP-адрес, на котором failover сервисы принимают failover-сообщения при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

Синтаксис

```
local-address { <ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no local-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.

Необходимый уровень привилегий

10

Командный режим

CONFIG-FAILOVER

Пример

```
rtt(config-failover)# local-address 192.168.1.1
```

multicast-address

Данной командой устанавливается многоадресный IP-адрес, который будет использоваться для обмена информации при работе резервирования failover-сервисов в multicast-режиме.

Использование отрицательной формы команды (no) удаляет многоадресный IP-адрес.

Синтаксис

```
multicast-address <ADDR>
no multicast-address
```

Параметры

<ADDR> – многоадресный IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-FAILOVER

Пример

```
rtt(config-failover)# multicast-address 238.0.0.10
```

multicast-group

Данной командой устанавливается идентификатор multicast-группы для обмена информацией при работе резервирования failover-сервисов в multicast-режиме.

Использование отрицательной формы команды (no) удаляет идентификатор группы.

Синтаксис

```
multicast-group <GROUP>
no multicast-group
```

Параметры

<GROUP> – multicast-группа, указывается в диапазоне [1000..9999].

Необходимый уровень привилегий

10

Командный режим

CONFIG-FAILOVER

Пример

```
rtt(config-failover)# multicast-group 1028
```

remote-address

Данной командой устанавливается IP-адрес, на который failover-сервисы отправляют failover-сообщения при работе в режиме резервирования.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес.

Синтаксис

```
remote-address { <ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no remote-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве local address.

Необходимый уровень привилегий

10

Командный режим

CONFIG-FAILOVER

Пример

```
rtt(config-failover)# remote-address 192.168.1.2
```

show high-availability state

Данная команда позволяет посмотреть общее состояние систем резервирования и роль устройства.

Синтаксис

```
show high-availability state
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show high-availability state
DHCP option 82 table:
    State:                          Disabled
    Last state change:              --
DHCP server:
VRF:                                --
    State:                          Successful synchronization
Firewall sessions and NAT translations:
    Tracking VRRP Group             1
    Tracking VRRP Group state:      Master
    State:                          Successful synchronization
    Fault Reason:                   --
    Last synchronization:           2022-11-25 13:19:04
```

vrrp-group

Данной командой указывается VRRP-группа, по состоянию которой будет определятся мастерство при работе failover-сервисов в режиме Active-Standby.

Использование отрицательной формы команды (no) удаляет указанную vrrp-group из конфигурации failover.

Синтаксис

```
vrrp-group <GRID>
no vrrp-group
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-FAILOVER

Пример

```
rtt(config-failover)# vrrp-group 2
```

23.6.2. Настройка резервирования DHCP

auto-partner-down

Данной командой устанавливается время отработки DHCP-lease, при котором аренда может быть продлена одним из узлов отказоустойчивости, не связываясь с другим.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ip dhcp-server failover auto-partner-down <TIME>
no ip dhcp-server failover auto-partner-down
```

Параметры

<TIME> – время, определяется в секундах [1..86400].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER-FAILOVER

Пример

```
rtt(config-dhcp-server-failover)# auto-partner-down 200
```

enable

Данной командой включается работа резервирования DHCP-сервера.

Использование отрицательной формы команды (no) выключает работа резервирования DHCP-сервера.

Синтаксис

```
[no] enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config-dhcp-server-failover)# enable
```

ip dhcp-server failover

Данной командой осуществляется переход в режим конфигурирования резервирования DHCP-сервера.

Использование отрицательной формы команды (no) удаляет настройки резервирования DHCP-сервера.

Синтаксис

```
[no] ip dhcp-server failover [ vrf <VRF> ] [ all ]
```

Параметры

<VRF> – имя VRF, задается строкой до 31 символа;

all – при использовании ключа при удалении, удаляется конфигурация резервирования DHCP-серверов во всех VRF и глобальной конфигурации.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-server failover
rtt(config-dhcp-server-failover)#
```

maximum-client-lead-time

Данной командой устанавливается максимальное время, на которое DHCP-сервер может продлить время аренды IP-адреса.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
maximum-client-lead-time <TIME>
no maximum-client-lead-time
```

Параметры

<TIME> – время, определяется в секундах [1..86400].

Значение по умолчанию

1800

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER-FAILOVER

Пример

```
rtt(config-dhcp-server-failover)# maximum-client-lead-time 200
```

mode

Данной командой настраивается режим работы DHCP failover.

Использование отрицательной формы команды (no) устанавливает режим работы DHCP failover по умолчанию.

Синтаксис

```
mode { active-active | active-standby }
no mode
```

Значение по умолчанию

Active-Active

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER-FAILOVER

Пример

```
rtt(config-dhcp-server-failover)# mode active-standby
```

role

Данной командой определяется роль DHCP-сервера при работе в режиме резервирования Active-Active.

Использование отрицательной формы команды (no) удаляет установленную роль DHCP-сервера при работе в режиме резервирования Active-Active.

Синтаксис

```
role <ROLE>
```

Параметры

<ROLE> – роль DHCP-сервера при работе в режиме резервирования:

- primary – режим активного DHCP-сервера;
- secondary – режим резервного DHCP-сервера.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER-FAILOVER

Пример

```
rtt(config-dhcp-server-failover)# role primary
```

show ip dhcp server failover

Данная команда позволяет посмотреть состояние резервирования DHCP-сервера.

Синтаксис

```
show ip dhcp server failover [ vrf <VRF> | all ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip dhcp server failover
VRF:       --
    State: Successful
```

23.6.3. Настройка резервирования Firewall/NAT

clear ip firewall failover counters

Данная команда позволяет очистить счетчики работы firewall failover.

Синтаксис

```
clear ip firewall failover counters [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip firewall failover counters
```

enable

Данной командой включается работа резервирования firewall.

Использование отрицательной формы команды (no) отключает резервирование firewall.

Синтаксис

```
[no] enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-FIREWALL-FAILOVER

Пример

```
rtt(config-firewall-failover)# enable
```

ip firewall failover

Данной командой осуществляется переход в режим конфигурирования резервирования сессий Firewall.

Использование отрицательной формы команды (no) удаляет настройки резервирования сессий Firewall.

Синтаксис

```
[no] ip firewall failover [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall failover
```

port

Данной командой устанавливается номер UDP-порта службы резервирования сессий Firewall, через который происходит обмен информацией при работе в unicast-режиме.

Использование отрицательной формы команды (no) удаляет номер порта службы резервирования сессий Firewall.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер порта службы резервирования сессий Firewall, указывается в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-FIREWALL-FAILOVER

Пример

```
rtt(config-firewall-failover)# port 3333
```

show ip firewall failover

Данная команда позволяет посмотреть состояние резервирования сессий Firewall.

Синтаксис

```
show ip firewall failover [ cache ] [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

cache – ключ для отображения счетчиков кэша.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример 1

```
rtt# show ip firewall failover
Communication interface:                    gigabitethernet 2/0/3.10
Status:                                     Running
Bytes sent:                                 2328
Bytes received:                             2388
Packets sent:                               148
Packets received:                           149
Send errors:                                0
Receive errors:                             0
Resend queue:
    Active entries:                         1
    Errors:
        No space left:                      0
Hold queue:
    Active entries:                         0
    Errors:
        No space left:                      0
```

Пример 2

```
rtt# show ip firewall failover cache
Internal sessions cache counters:
    Active entries:                         1
    Added:                                  2
    Deleted:                                1
    Updated:                                0
    Failed adding:                          0
        No memory left:                     0
        No space left:                      0
    Failed deleting:                        0
        No entry found:                     0
    Failed updating:                        0
        No entry found:                     0
External sessions cache counters:
    Active entries:                         1
    Added:                                  1
    Deleted:                                0
    Updated:                                0
    Installed to Kernel:                    0
    Failed adding:                          0
        No memory left:                     0
```

```
        No space left:                  0
    Failed deleting:                    0
        No entry found:                 0
    Failed updating:                    0
        No entry found:                 0
    Failed installing to Kernel:        0
```

show ip firewall sessions

Данная команда используется для просмотра кэшей IP-сессий при работе firewall failover.

Синтаксис

```
show ip firewall sessions [ configuration ] [ tracking ] [ inside-destination-address <ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-address <ADDR> ] [ outside-source-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ summary ] [ protocol <TYPE> ] [ expected ] [ failover ]  [ vrf <VRF> ]
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

configuration – ключ для отображения конфигурации;

tracking – ключ для отображения протоколов отслеживания;

failover – ключ для отображения кэша отказоустойчивых сессий;

inside-source-address – команда для указания IP-адреса источника приходящих пакетов;

inside-destination-address – команда для указания IP-адреса назначения приходящих пакетов;

outiside-source-address – команда для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

expected – выводит информацию по ожидаемым сессиям;

summary – выводит суммарную статистику по IP-сессиям.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip firewall sessions
 Virtual router   Virtual IP                                  Priority   Preemption   State    Inherit   Sync group ID
 --------------   -----------------------------------------   --------   ----------   ------   -------   -------------
 1                fe80::1/112, fc00::254/112                  100        Enabled      Master   --        --
```

show ip firewall sessions failover

Данная команда используется для просмотра кэшей IP-сессий при работе firewall failover.

Синтаксис

```
show ip firewall sessions failover { internal | external } [ protocol <TYPE> ] [ inside-source-address <ADDR>] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ summary ] [ configuration ] [ expected ] [ vrf <VRF> ]
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа;

internal – кэш синхронизации с ядром.

external – кэш синхронизации с соседом.

inside-source-address – команда для указания IP-адреса источника приходящих пакетов;

inside-destination-address – команда для указания IP-адреса назначения приходящих пакетов;

outiside-source-address – команда для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

expected – выводит информацию по ожидаемым сессиям;

summary – выводит суммарную статистику по IP-сессиям.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip firewall sessions failover internal
  Codes: E - expected, U - unreplied,
         A - assured, C - confirmed

 Prot    Aging       Inside source          Inside destination     Outside source         Outside destination    Pkts        Bytes       Status
 -----   ----------  ---------------------  ---------------------  ---------------------  ---------------------  ----------  ----------  ------
 udp     29          10.10.10.2:500         10.10.10.1:500         10.10.10.2:500         10.10.10.1:500          --          --          UC
```

show ip nat translations failover

Данная команда используется для просмотра кэша IP-сессий трансляции при работе nat failover.

Синтаксис

```
show ip nat translations failover { internal | external } [ protocol <TYPE> ] [ inside-source-address <ADDR> ] [ outiside-source-address <ADDR> ] [ inside-destination-address <ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ summary ] [ vrf <VRF> ]
```

Параметры

summary – выводит суммарную статистику по сессиям трансляции;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Для Source NAT:

- inside-source-address – ключ для указания IP-адреса источника до трансляции;
- inside-destination-address – ключ для указания IP-адреса назначения на входе в маршрутизатор;
- outiside-source-address – ключ для указания IP-адреса источника после трансляции;
- outside-destination-address – ключ для указания IP-адреса назначения на выходе из маршрутизатора.
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Для Destination NAT:

- inside-source-address – ключ для указания IP-адреса источника на выходе из маршрутизатора;
- inside-destination-address – ключ для указания IP-адреса назначения после трансляции;
- outiside-source-address – ключ для указания IP-адреса источника на входе в маршрутизатор;
- outside-destination-address – ключ для указания IP-адреса назначения до трансляции;
- inside-source-port – ключ для указания TCP/UDP-порта отправителя до трансляции;
- outside-source-port – ключ для указания TCP/UDP-порта отправителя после трансляции;
- inside-destination-port – ключ для указания TCP/UDP-порта назначения до трансляции;
- outside-destination-port – ключ для указания TCP/UDP-порта назначения после трансляции.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip nat translations  failover internal
Prot   Inside source          Inside destination     Outside source         Outside destination    Pkts
----   --------------------   --------------------   --------------------   --------------------   ----------
tcp    192.168.10.10:39268    217.134.15.73:22       222.222.222.2:39268    217.134.15.73:22       --
```

sync-type

Данной командой определяется режим обмена информацией между основным и резервным маршрутизаторами.

Использование отрицательной формы команды (no) удаляет режим работы резервирования Firewall.

Синтаксис

```
sync-type <MODE>
no sync-type
```

Параметры

<MODE> – режим обмена информацией:

- unicast – режим unicast;
- multicast – режим multicast.

Необходимый уровень привилегий

10

Командный режим

CONFIG-FIREWALL-FAILOVER

Пример

```
rtt(config-firewall-failover)# sync-type multicast
```

23.6.4. Настройка Crypto-Sync

crypto-sync

Данной командой осуществляется переход в режим конфигурирования резервирования сертификатов.

Использование отрицательной формы команды (no) удаляет настройки резервирования сертификатов.

Синтаксис

```
[no] crypto-sync
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# crypto-sync
rtt(config-crypto-sync)#
```

enable

Данной командой включается функция синхронизации сертификатов.

Использование отрицательной формы команды (no) отключает функцию.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CRYPTO-SYNC

Пример

```
rtt(config-crypto-sync)# enable
```

remote-delete

Данной командой включается функция удаления сертификатов при синхронизации.

Использование отрицательной формы команды (no) выключает функционал.

Синтаксис

```
remote-delete
no remote-delete
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CRYPTO-SYNC

Пример

```
rtt(config-crypto-sync)# remote-delete
```

show crypto-sync

Данная команда позволяет посмотреть состояние резервирования сертификатов.

Синтаксис

```
show crypto-sync
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show crypto-sync
    Role:                       Master
    State:                      Synchronized
    Last synchronization:       2025-11-17 10:09:56
```

sync crypto force

Данной командой запускается полная синхронизация файлов в ручном режиме.

Синтаксис

```
sync crypto force
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# sync crypto force
```

24. УПРАВЛЕНИЕ КЛАСТЕРОМ

cluster

Данной командой осуществляется переход в режим конфигурирования кластера.

Использование отрицательной формы команды (no) удаляет настройки кластера.

Синтаксис

```
[no] cluster
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# cluster
rtt(config-cluster)#
```

cluster-interface

Данной командой устанавливается интерфейс, через который будет происходить обмен служебными сообщениями между юнитами в кластере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
cluster-interface bridge [<BRIDGE-ID>]
no cluster-interface
```

В текущей версии в качестве интерфейса можно выбрать только bridge.

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLUSTER

Пример

```
rtt(config-cluster)# cluster-interface bridge 1
```

enable

Данной командой включается кластер.

Использование отрицательной формы команды (no) выключает кластер.

Синтаксис

```
[no] enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLUSTER

Пример

```
rtt(config-cluster)# enable
```

mac-address

Данная команда позволяет задать MAC-адрес для определенного юнита.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mac-address <ADDR>
no mac-address
```

Параметры

<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLUSTER-UNIT

Пример

```
rtt(config-cluster-unit)# mac-address A8:F9:B0:00:00:04
```

show cluster status

Данная команда позволяет посмотреть состояние юнитов в кластере.

Синтаксис

```
show cluster status
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cluster status
Unit   Hostname              Role         MAC address         State          IP address
----   ------------------   ----------   -----------------   ------------   ----------
```

```
1*      rtt                   Active       cc:9d:a2:71:b6:b7   Joined         10.0.0.1
2       rtt                   Standby      cc:9d:a2:71:b8:ca   Joined         10.0.0.2
```

show cluster sync configuration

Данная команда позволяет посмотреть состояние синхронизации различных подсистем в кластере.

Синтаксис

```
show cluster sync configuration
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cluster sync configuration
Unit   Hostname               Config        Date
----   --------------------   -----------   -----------
1      rtt                    Enabled       Enabled
2*     rtt                    Enabled       Enabled
```

show cluster sync status

Данная команда позволяет посмотреть состояние синхронизации различных подсистем в кластере.

Синтаксис

```
show cluster sync status
```

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show cluster sync status
System part              Synced
----------------------   ------
candidate-config         Yes
running-config           Yes
SW version               Yes
licence                  Yes
```

```
licence (After reboot)    Yes
date                      Yes
```

show cluster-unit-licences

Данная команда позволяет посмотреть загруженные лицензии для устройств в кластере.

Синтаксис

```
show cluster-unit-licences
```

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show cluster-unit-licences
Serial number     Features
---------------   ------------------------------------------------------------
NP18000582        BRAS,IPS,WIFI
NP18000622        BRAS,IPS,WIFI
```

sync cluster system force

Данная команда вызывает процесс принудительной синхронизации версии ПО, конфигурации, лицензии и времени выбранных юнитов в кластере.

Синтаксис

```
sync cluster system force [ unit { <RANGE-ID> | local } ]
```

Параметры

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – подставляет номер локального юнита.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# sync cluster system force
```

sync config disable

Данная команда отключает синхронизацию конфигураций в кластере между юнитами.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] sync config disable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLUSTER

Пример

```
rtt(config-cluster)# sync config disable
```

unit

Данной командой осуществляется переход в режим конфигурирования юнита в кластере.

Использование отрицательной формы команды (no) удаляет юнит из кластера.

Синтаксис

```
[no] unit <ID>
```

Параметры

<ID> – номер юнита, принимает значения [1..4].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLUSTER

Пример

```
rtt(config-cluster)# unit 1
rtt(config-cluster-unit)#
```

25. БЕЗОПАСНОСТЬ

25.1. Управление списками контроля доступа (ACL)

action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
action <ACT>
no action
```

Параметры

<ACT> – назначаемое действие:

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# action permit
```

description

Данная команда используется для изменения описания конфигурируемого списка контроля доступа.

Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание списка контроля доступа, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL

Пример

```
rtt(config-acl)# description "Drop SSH traffic"
```

enable

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Правило выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# enable
```

ip access-list extended

Данная команда используется для создания списка контроля доступа и перехода в режим конфигурирования списка.

Использование отрицательной формы команды (no) удаляет заданный список контроля доступа.

Синтаксис

```
[no] ip access-list extended <NAME>
```

Параметры

<NAME> – имя создаваемого списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip access-list extended acl-ssh-drop
rtt(config-acl)#
```

match cos

Данной командой устанавливается значение 802.1p приоритета, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match cos <COS>
no match cos
```

Параметры

<COS> – значение 802.1p приоритета, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match cos 2
```

match destination-address

Данной командой устанавливаются IP-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match destination-address { <ADDR> <MASK> | any }
no match destination-address
```

Параметры

<ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MASK> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match destination-address 10.10.10.0 255.255.255.0
```

match destination-mac

Данной командой устанавливаются MAC-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match destination-mac <ADDR> <WILDCARD>
no match destination-mac
```

Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match destination-mac A8:F9:4B:AA:00:41 00:00:00:00:00:FF
```

match destination-port

Данной командой устанавливается номер TCP/UDP-порта получателя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
match destination-port <TYPE> {<PORT> | <PORT>-<PORT>}
no match destination-port
```

Параметры

<TYPE> – принимает значения «any или «port-range»;

Можно указать один порт либо указать диапазон портов через «-»;

<PORT> – продолжение команды port-range, номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match destination-port port-range 22
```

match dscp

Данной командой устанавливается значение кода DSCP, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match dscp <DSCP>
no match dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match dscp 55
```

match ip-precedence

Данной командой устанавливается значение кода IP Precedence, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match ip-precedence <IPP>
no match ip-precedence
```

Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match ip-precedence 5
```

match protocol

Данной командой устанавливается имя IP-протокола, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match protocol <TYPE>
no match protocol
match protocol-id <ID>
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

При указании значения «any» правило будет срабатывать для любого протокола.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match protocol tcp
```

match source-address

Данной командой устанавливаются IP-адреса отправителя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match source-address { <ADDR> <MASK> | any }
no match source-address
```

Параметры

<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<MASK> – маска IP-адреса, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Биты маски, установленные в 0, задают биты IP-адреса, исключаемые из сравнения при поиске.

При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match source-address 10.100.100.0 255.255.255.0
```

match source-mac

Данной командой устанавливаются MAC-адреса отправителя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match source-mac <ADDR> <WILDCARD>
no match source-mac
```

Параметры

<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<WILDCARD> – маска MAC-адреса, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF]. Биты маски, установленные в 0, задают биты MAC-адреса, исключаемые из сравнения при поиске.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match source-mac A8:F9:4B:AA:00:40 00:00:00:FF:FF:FF
```

match source-port

Данной командой устанавливается номер TCP/UDP-порта отправителя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match source-port <TYPE> { <PORT> | <PORT>-<PORT> }
no match source-port
```

Параметры

<TYPE> – принимает значения «any» или «port-range»;

Можно указать один порт либо указать диапазон портов через «-»;

<PORT> – Продолжение команды port-range, номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match source-port any
```

match vlan

Данной командой устанавливается значение идентификационного номера VLAN, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match vlan <VID>
no match vlan
```

Параметры

<VID> – идентификационный номер VLAN, принимает значения [1…4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL-RULE

Пример

```
rtt(config-acl-rule)# match vlan 100
```

rule

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-ACL-RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..4094].

Необходимый уровень привилегий

10

Командный режим

CONFIG-ACL

Пример

```
rtt(config-acl)# rule 10
rtt(config-acl-rule)#
```

service-acl input

Данная команда используется для привязки указанного списка контроля доступа к конфигурируемому интерфейсу для фильтрации входящего трафика.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному интерфейсу.

Использование фильтрации пакетов при помощи ACL на физическом интерфейсе увеличивает нагрузку на CPU маршрутизатора.

Для фильтрации трафика рекомендуется использовать функционал ZoneBased Firewall.

Синтаксис

```
service-acl input <NAME>
no service-acl input
```

Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# service-acl input acl-ssh-drop
```

service-acl output

Данная команда используется для привязки указанного списка контроля доступа к конфигурируемому интерфейсу для фильтрации исходящего трафика.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному интерфейсу.

Использование фильтрации пакетов при помощи ACL на физическом интерфейсе увеличивает нагрузку на CPU маршрутизатора.

Для фильтрации трафика рекомендуется использовать функционал ZoneBased Firewall.

Синтаксис

```
service-acl output <NAME>
no service-acl output
```

Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-PORT-CHANNEL

Пример

```
rtt(config-if-gi)# service-acl output acl-ftp
```

show ip access-list

Данная команда используется для просмотра списков управления доступом.

Синтаксис

```
show ip access-list [ <NAME> [ <ORDER> ] ]
```

Параметры

<NAME> – имя списка управления доступом, задаётся строкой до 31 символа;

<ORDER> – номер правила, принимает значения [1..4096]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip access-list
Name                              Description
--------------------------------  ----------------------------------------
acl-telnet-drop                   --
acl-ssh-drop                      Drop SSH traffic
rtt# show ip access-list acl-ssh-drop
Index:                        1
Matching pattern:
    Protocol:                 TCP(6)
    Source MAC address:       any
    Source IP address:        any
    Source port:              any
    Destination MAC address:  any
    Destination IP address:   any
    Destination port:         22
Action:                       Deny
Status:                       Enabled
------------------------------------------------------------------------------
Index:                        2
Matching pattern:
    Protocol:                 any
    Source MAC address:       any
    Source IP address:        any
    Destination MAC address:  any
    Destination IP address:   any
Action:                       Permit
Status:                       Enabled
------------------------------------------------------------------------------
```

25.2. Управление Firewall

action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
action { permit | deny | reject | netflow-sample | sflow-sample | rate-limit total pps <RATE> | session-limit [total <SESSION>] [per-source-host <SESSION>] } [log]
no action
```

Параметры

- permit – прохождение трафика разрешается;
- deny – прохождение трафика запрещается;
- reject – прохождение трафика запрещается, а также посылается отправителю ответ об ошибке;
- netflow-sample – прохождение трафика разрешается, осуществляется экспорт статистики по протоколу Netflow;
- sflow-sample – прохождение трафика разрешается, осуществляется экспорт статистики по протоколу sFlow;
- rate-limit total pps <RATE> – прохождение трафика разрешается, ограничивается количество обрабатываемых правилом пакетов в секунду. Команда применима только для наборов правил в зону self:
    - <RATE> – количество пакетов в секунду, принимает значения [1..10000].

Значение по умолчанию

Действие не настроено, логирование отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# action permit
```

check output-interface

Данной командой разрешается очистка Firewall-сессий, в случае если выходной интерфейс изменен.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
check output-interface
no check output-interface
```

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# check output-interface
```

clear ip firewall counters

Данной командой осуществляется сброс счетчиков правил Firewall.

Синтаксис

```
clear ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip firewall counters trusted self
```

clear ip firewall sessions

Данной командой осуществляется удаление активных IP-сессий.

Синтаксис

```
clear ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address <ADDR> ] [ outiside-source-address <ADDR> ] [ inside-destination-address <ADDR> ] [ outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – ключ для указания IP-адреса источника приходящих пакетов;

inside-destination-address – ключ для указания IP-адреса назначения приходящих пакетов;

outiside-source-address – ключ для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – ключ для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip firewall sessions vrf VRF1
```

clear ipv6 firewall counters

Данной командой осуществляется сброс счетчиков правил Firewall.

Синтаксис

```
clear ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE>
[<ORDER>] ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут очищены счетчики правил в указанном VRF;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будут очищены счетчики только по данному правилу.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ipv6 firewall counters trusted self
```

clear ipv6 firewall sessions

Данной командой осуществляется удаление активных IPv6-сессий.

Синтаксис

```
clear ipv6 firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-
address <IPV6-ADDR> ] [ outiside-source-address <IPV6-ADDR> ] [ inside-destination-
address <IPV6-ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-
port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [
outside-destination-port <PORT> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут удалены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IPv6-адреса источника приходящих пакетов;

inside-destination-address – команда для указания IPv6-адреса назначения приходящих пакетов;

outiside-source-address – команда для указания IPv6-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IPv6-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ipv6 firewall sessions vrf VRF1
```

description

Данная команда используется для изменения описания конфигурируемой зоны или пары зон безопасности. Использование отрицательной формы команды (no) удаляет установленное описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание зоны безопасности, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE

CONFIG-SECURITY-ZONE-PAIR

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone)# description "Trusted interfaces"
```

enable

Данная команда используется для активирования правила.

Использование отрицательной формы команды (no) деактивирует правило.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-rule)# enable
```

ip firewall disable

Данная команда используется для отключения функции Firewall глобально на всех сетевых интерфейсах.

Использование отрицательной формы команды (no) включает функцию Firewall глобально.

Синтаксис

```
[no] ip firewall disable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall disable
```

ip firewall disable

Данная команда используется для отключения функции Firewall на сетевом интерфейсе.

Использование отрицательной формы команды (no) включает функцию Firewall на сетевом интерфейсе.

Синтаксис

```
[no] ip firewall disable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-CELLULAR-MODEM

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-L2TP

CONFIG-LT

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-OPENVPN

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-if-gi)# ip firewall disable
```

ip firewall mode

Данная команда используется для выбора режима работы межсетевого экрана.

Использование отрицательной формы команды (no) устанавливает режим работы межсетевого экрана по умолчанию.

Синтаксис

```
ip firewall mode <MODE>
no ip firewall mode
```

Параметры

<MODE> – режим работы межсетевого экрана, может принимать значения:

- stateful – режим, при котором маршрутизатор отслеживает сессии. Первые пакеты сессии проходят полный цикл проверки согласно правилам межсетевого экрана, а последующие пакеты сессии маршрутизируются без дополнительных проверок. В этом режиме, если "прямой" трафик разрешён, "ответный" трафик разрешается автоматически. Данное правило не распространяется на работу механизма DPI.
- stateless – режим, при котором маршрутизатор не отслеживает сессии. Каждый пакет проходит полный цикл проверки согласно правилам межсетевого экрана, что существенно снижает производительность оборудования. Использование данного режима допустимо только в условиях крайней необходимости.

Значение по умолчанию

stateful

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall mode stateless
```

ip firewall sessions counters

Командой выполняется включение счетчиков сессий для NAT и Firewall. Счетчики увеличиваются только тогда, когда устанавливается новая сессия. Для установленных сессий увеличения значений счетчиков не происходит при прохождении пакетов. Включение счетчиков снижает производительность маршрутизатора.

Команды для просмотра счетчиков и сессий описаны в разделах show ip firewall counters, show ip firewall sessions, show ipv6 firewall counters и show ipv6 firewall sessions.

Использование отрицательной формы команды (no) отключает счетчики сессий.

Синтаксис

```
[no] ip firewall sessions counters
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions counters
```

ip firewall sessions generic-timeout

Данной командой определяется время жизни сессии для неподдерживаемых протоколов, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions generic-timeout <TIME>
no ip firewall sessions generic-timeout
```

Параметры

<TIME> – время жизни сессии для неподдерживаемых протоколов, принимает значения в секундах [1..8553600].

Значение по умолчанию

60 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions generic-timeout 60
```

ip firewall sessions icmp-timeout

Данной командой определяется время жизни ICMP-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions icmp-timeout <TIME>
no ip firewall sessions icmp-timeout
```

Параметры

<TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions icmp-timeout 60
```

ip firewall sessions icmpv6-timeout

Данной командой определяется время жизни ICMPv6-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions icmpv6-timeout <TIME>
no ip firewall sessions icmpv6-timeout
```

Параметры

<TIME> – время жизни ICMPv6-сессии, принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions icmpv6-timeout 60
```

ip firewall sessions max-expect

Данной командой определяется размер таблицы сессий, ожидающих обработки.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions max-expect <COUNT>
no ip firewall sessions max-expect
```

Параметры

<COUNT> – размер таблицы, принимает следующие значения:

- R800 – 3130000;
- R200 – 2259000;
- R100 – 1574000.

Значение по умолчанию

256

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions max-expect 512
```

ip firewall sessions max-tracking

Данной командой определяется размер таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions max-tracking <COUNT>
no ip firewall sessions max- tracking
```

Параметры

<COUNT> – размер таблицы, принимает следующие значения:

- R800 – 3130000;
- R200 – 2259000;
- R100 – 1574000.

Значение по умолчанию

512000

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions max-tracking 256000
```

ip firewall sessions tcp-connect-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение устанавливается», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions tcp-connect-timeout <TIME>
no ip firewall sessions tcp-connect-timeout
```

Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение устанавливается», принимает значения в секундах [1..8553600].

Значение по умолчанию

60 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions tcp-connect-timeout 120
```

ip firewall sessions tcp-disconnect-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение закрывается», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions tcp-disconnect-timeout <TIME>
no ip firewall sessions tcp-disconnect-timeout
```

Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение закрывается», принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions tcp-disconnect-timeout 10
```

ip firewall sessions tcp-estabilished-timeout

Данной командой определяется время жизни TCP-сессии в состоянии «соединение установлено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions tcp-estabilished-timeout <TIME>
no ip firewall sessions tcp-estabilished-timeout
```

Параметры

<TIME> – время жизни TCP-сессии в состоянии «соединение установлено», принимает значения в секундах [1..8553600].

Значение по умолчанию

120 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions tcp-estabilished-timeout 3600
```

ip firewall sessions tcp-latecome-timeout

Данной командой определяется время ожидания, по истечении которого происходит фактическое удаление закрытой TCP-сессии из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions tcp-latecome-timeout <TIME>
no ip firewall sessions tcp-latecome-timeout
```

Параметры

<TIME> – время ожидания, принимает значения в секундах [1..8553600].

Значение по умолчанию

120 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions tcp-latecome-timeout 10
```

ip firewall sessions tracking

Данной командой включается функция отслеживания сессий уровня приложений для отдельных протоколов.

Использование отрицательной формы команды (no) отключает функцию отслеживания сессий уровня приложений для отдельных протоколов.

Синтаксис

```
ip firewall sessions tracking { <PROTOCOL> |  sip [ port <OBJECT-GROUP-SERVICE> |
session-lifetime <TIME> ] }
no ip firewall sessions tracking { <PROTOCOL> |  sip [ port <OBJECT-GROUP-SERVICE>
] | all }
```

Параметры

<PROTOCOL> – протокол уровня приложений, сессии которого должны отслеживаться, принимает значения [ftp, h323, pptp, netbios-ns];

<OBJECT-GROUP-SERVICE> – имя профиля TCP/UDP-портов SIP-сессии, задаётся строкой до 31 символа. Если группа не указана, то отслеживание сессий SIP будет осуществляться для порта 5060;

Вместо имени отдельного протокола можно использовать ключ "all", который включает функцию отслеживания сессий уровня приложений для всех доступных протоколов.

<TIME> – время жизни отслеживаемой SIP-сессии, по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий. Принимает значения в секундах [1..8553600].

Значение по умолчанию

Отключено для всех протоколов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions tracking ftp
```

ip firewall sessions udp-assured-timeout

Данной командой определяется время жизни UDP-сессии в состоянии «соединение подтверждено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions udp-assured-timeout <TIME>
no ip firewall sessions udp-assured-timeout
```

Параметры

<TIME> – время жизни UDP-сессии в состоянии «соединение подтверждено», принимает значения в секундах [1..8553600].

Значение по умолчанию

180 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions udp-assured-timeout 3600
```

ip firewall sessions udp-wait-timeout

Данной командой определяется время жизни UDP-сессии в состоянии «соединение не подтверждено», по истечении которого она считается устаревшей и удаляется из таблицы отслеживаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip firewall sessions udp-wait-timeout <TIME>
no ip firewall sessions udp-wait-timeout
```

Параметры

<TIME> – время жизни UDP-сессии в состоянии «соединение не подтверждено», принимает значения в секундах [1..8553600].

Значение по умолчанию

30 секунд.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions udp-wait-timeout 60
```

ip firewall sessions unknown

Данной командой настраивается фильтрация пакетов, для которых не удалось определить принадлежность к какому-либо известному соединению и которые не являются началом нового соединения.

Использование отрицательной формы команды (no) включает отбрасывание всех неизвестных сессий.

Синтаксис

```
ip firewall sessions unknown <ACTION>
[no] ip firewall sessions unknown
```

Параметры

<ACTION> – правило обработки неизвестных сессий для межсетевого экрана:

- permit – разрешить неизвестные сессии;
- deny – отбрасывать неизвестные сессии;
- reject – отбрасывать неизвестные сессии и отправлять обратно пакет с ошибкой.

Значение по умолчанию

reject

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions unknown permit
```

match application

Данной командой устанавливается профиль приложений, для которых должно срабатывать правило. Данная функция используется для фильтрации по приложениям (механизм DPI).

При использовании параметра «not» правило будет срабатывать для приложений, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] application <OBJ-GROUP-APPLICATION>
no match application
```

Параметры

<OBJ-GROUP-APPLICATION> – имя профиля приложений, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match application APP_DENY
```

match destination-address

Данной командой устанавливается профиль IP-адресов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для IP-адресов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-address { address-range { <ADDR>[-<ADDR>] |
<IPV6-ADDR>[-<IPV6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } | object-
group { network <OBJ-GROUP-NETWORK-NAME> |
domain-name <OBJ-GROUP-DOMAIN-NAME> } | any }
no match destination-address
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;

object-group domain-name <OBJ-GROUP-DOMAIN-NAME> – имя профиля доменных имен, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-pair-zone-rule)# match destination-address object-group network
remote_workspace
```

match destination-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для связок IP-адресов и TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-address-port { address-port { <ADDR>:<PORT> | <IPV6-
ADDR>:<PORT> } | object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }
no match destination-address-port
```

Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255], номер порта, принимает значение [1..65535]. <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF], номер порта, принимает значение [1..65535];

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match destination-address object-group local
```

match destination-mac

Данной командой устанавливается MAC-адрес получателя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов получателя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-mac { mac <ADDR> | object-group <OBJ-GROUP-MAC-NAME> }
no match destination-mac
```

Параметры

<ADDR> – MAC-адрес получателя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

object-group <OBJ-GROUP-MAC-NAME> – имя профиля MAC-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого MAC-адреса получателя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match destination-mac mac A8:F9:4B:AA:00:40
```

match destination-nat

Данной командой устанавливается ограничение, при котором правило будет срабатывать только для трафика, измененного сервисом трансляции IP-адресов и портов получателя.

При использовании параметра «not» правило будет срабатывать для трафика, не измененного сервисом трансляции IP-адресов и портов получателя. Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] destination-nat
no match destination-nat
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match destination-nat
```

match destination-port

Данной командой устанавливается профиль TCP/UDP-портов получателя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов получателя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
match [not] destination-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME> | any }
no match destination-port
```

Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

<PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match destination-port object-group ssh
```

match fragment

Данной командой определяются фрагментированные пакеты, направленные на устройство. Команда применима только в правилах между зонами any self. Под действие правила попадают второй и последующие фрагменты пакета. Обработка пакетов этим правилом происходит до трансляции адресов DNAT.

При использовании параметра «not» правило будет срабатывать для нефрагментированных пакетов.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] fragment
no match fragmen
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match fragment
```

match icmp

Данная команда используется для настройки параметров протокола ICMP, если он выбран командой *match protocol*. Данной командой устанавливается тип и код сообщений протокола ICMP, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP, кроме указанных.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] icmp { <ICMP_TYPE> <ICMP_CODE> | <OPTION> }
no match icmp
```

Параметры

<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255];

<ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP;

<OPTION> – стандартные типы ICMP-сообщений, могут принимать значения:

- administratively-prohibited;
- alternate-address;
- conversion-error;
- dod-host-prohibited;
- dod-network-prohibited;
- echo;
- echo-reply;
- host-isolated;
- host-precedence;
- host-redirect;
- host-tos-redirect;
- host-tos-unreachable;
- host-unknown;
- host-unreachable;
- information-reply;
- information-request;
- mask-reply;
- mask-request;
- network-redirect;
- network-tos-redirect;
- network-tos-unreachable;
- network-unknown;
- network-unreachable;
- option-missing;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- precedence;
- protocol-unreachable;
- reassembly-timeout;
- router-advertisement;
- router-solicitation;
- source-quench;
- source-route-failed;
- time-exceeded;
- timestamp-reply;

- timestamp-request;
- traceroute.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match icmp 2 any
```

match icmp6

Данная команда используется для настройки параметров протокола ICMP6, если он выбран командой *match protocol*. Данной командой устанавливается тип и код сообщений протокола ICMP6, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех типов и кодов сообщений протокола ICMP6, кроме указанных.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] icmp6 { <ICMP6_TYPE> <ICMP6_CODE> | <OPTION> }
no match icmp6
```

Параметры

<ICMP6_TYPE> – тип сообщения протокола ICMP6, принимает значения [0..255];

<ICMP6_CODE> – код сообщения протокола ICMP6, принимает значения [0..255]. При указании значения «any» правило будет срабатывать для любого кода сообщения протокола ICMP6;

<OPTION> – стандартные типы ICMP6-сообщений, могут принимать значения:

- address-unreachable;
- beyond-source-address;
- echo;
- echo-reply;
- error-in-p-route;
- error-source-routing-header;
- headers-too-long;
- multicast-listener-done;
- multicast-listener-query;
- multicast-listener-report;

- neighbor-advertisement;
- neighbor-solicitation;
- no-communication;
- no-route;
- packet-too-big;
- parameter-problem;
- port-unreachable;
- reassembly-timeout;
- redirect;
- reject-route;
- router-advertisement;
- router-solicitation;
- time-exceeded.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match icmp6 2 any
```

match ip-option

Данной командой определяются пакеты, содержащие опции в IP-заголовках. Команда применима только в правилах между зонами any self.

При использовании параметра «not» правило будет срабатывать для пакетов, не содержащих опций в IP-заголовках.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] ip-option
no match ip-option
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match ip-options
```

match protocol

Данной командой устанавливается имя или номер IP-протокола, для которого должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для всех протоколов, кроме указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] protocol <TYPE>
no match protocol
match [not] protocol-id <ID>
no match protocol-id
```

Параметры

<TYPE> – тип протокола, принимает значения: esp, icmp, icmp6, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rsvp, l2tp, gre.

При указании значения «any» правило будет срабатывать для любых протоколов;

<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match protocol udp
```

match source-address

Данной командой устанавливается профиль IP-адресов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для IP-адресов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-address { address-range { <ADDR>[-<ADDR>] | <IPV6-ADDR>[-<IPV6-ADDR>] } | prefix { <ADDR/LEN> | <IPv6-ADDR/LEN> } |
object-group { network <OBJ-GROUP-NETWORK-NAME> | domain-name <OBJ-GROUP-DOMAIN-NAME> } | any }

no match source-address
```

Параметры

address-range <ADDR>[-<ADDR>] – диапазон IP-адресов для правил firewall. Если не указывать IP-адрес конца диапазона, то в качестве IP-адреса для срабатывания правила используется только IP-адрес начала диапазона.
Параметр задаётся в виде A.B.C.D, где каждая часть принимает значения [0..255]; <IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

prefix <ADDR/LEN> – IP-подсеть, используемая для срабатывания правила фильтрации firewall. Параметр задаётся в виде A.B.C.D/E, где каждая часть A – D принимает значения [0..255] и E принимает значения [1..32]; <IPv6-ADDR/LEN> – IPv6-адрес, задаётся в виде X:X:X:X::X/E, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и E принимает значения [1..128];

object-group network <OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа;

object-group domain-name <OBJ-GROUP-DOMAIN-NAME> – имя профиля доменных имен, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match source-address object-group network
remote
```

match source-address-port

Данной командой устанавливается профиль связок IP-адресов и TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для связок IP-адресов и TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-address-port { address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT>
} |
object-group <OBJ-GROUP-ADDRESS-PORT-NAME> | any }

no match source-address-port
```

Параметры

address-port { <ADDR>:<PORT> | <IPV6-ADDR>:<PORT> } – связка IP-адресов и TCP/UDP-портов; IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; номер порта, принимает значение [1..65535]; <IPV6-ADDR>:<PORT> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]; номер порта, принимает значение [1..65535];

object-group <OBJ-GROUP-ADDRESS-PORT-NAME> – имя профиля связок IP-адресов и TCP/UDP-портов, задаётся строкой до 31 символа.

При указании значения «any» правило не будет учитывать данный способ фильтрации.

Значение по умолчанию

any

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match source-address-port object-group admin
```

match source-mac

Данной командой устанавливается MAC-адрес отправителя, для которого должно срабатывать правило.

При использовании параметра «not» (match not) правило будет срабатывать для MAC-адресов отправителя, отличных от указанного.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-mac { mac <ADDR> | object-group <OBJ-GROUP-MAC-NAME> }
no match source-mac
```

Параметры

mac <ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

object-group <OBJ-GROUP-MAC-NAME> – имя профиля MAC-адресов, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого MAC-адреса источника.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match source-mac mac A8:F9:4B:AA:00:40
```

match source-port

Данной командой устанавливается профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило.

При использовании параметра «not» правило будет срабатывать для TCP/UDP-портов отправителя, которые не входят в указанный профиль.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
match [not] source-port { port-range <PORT>[-<PORT>] | object-group <PORT-SET-NAME>
| any }
no match source-port
```

Параметры

port-range <PORT>[-<PORT>] – address-port <PORT>[-<PORT>] – диапазон TCP/UDP-портов для правил firewall. Если не указывать TCP/UDP-порт конца диапазона, то в качестве TCP/UDP-порта для срабатывания правила используется только порт начала диапазона.

object-group <PORT-SET-NAME> – имя профиля TCP/UDP-портов, задаётся строка до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR-RULE

Пример

```
rtt(config-security-zone-pair-rule)# match source-port object-group telnet
```

ports firewall enable

Данная команда активирует фильтрацию и режим отслеживания сессий при прохождении пакетов между членами Bridge-интерфейса.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
[no] ports firewall enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-BRIDGE

Пример

```
rtt(config-bridge)# ports firewall enable
```

rearrange

Данная команда меняет шаг между созданными правилами.

Синтаксис

```
rearrange <VALUE>
```

Параметры

<VALUE> – шаг между правилами, принимает значения [1..50].

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR

Пример

```
rtt(config-security-zone-pair)# rearrange 10
```

renumber rule

Данная команда меняет номер правила.

Синтаксис

```
renumber rule <CUR_ORDER> <NEW_ORDER>
```

Параметры

<CUR_ORDER> – текущий номер правила, принимает значения [1..10000];

<NEW_ORDER> – новый номер правила, принимает значения [1..10000].

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR

Пример

```
rtt(config-security-zone-pair)# renumber rule 13 100
```

rule

Данная команда используется для создания правила и перехода в командный режим SECURITY ZONE PAIR RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..10000]. Если при удалении используется значение параметра "all", то будут удалены все правила для конфигурируемой пары зон безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SECURITY-ZONE-PAIR

Пример

```
rtt(config-security-zone-pair)# rule 10
rtt(config-security-zone-pair-rule)#
```

security zone

Данная команда используется для создания зон безопасности и перехода в режим конфигурирования зоны.

Использование отрицательной формы команды (no) удаляет заданную зону безопасности.

Синтаксис

```
[no] security zone [ <NAME> | all ]
```

Параметры

<NAME> – имя создаваемой зоны безопасности, задаётся строкой до 12 символов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все зоны безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security zone trusted
rtt(config-security-zone)#
```

security-zone

Данная команда используется для добавления выбранного сетевого интерфейса в зону безопасности. Использование отрицательной формы команды (no) удаляет интерфейс из зоны.

Синтаксис

```
security-zone <NAME>
no security-zone
```

Параметры

<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-LT

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-OPENVPN

CONFIG-L2TP-SERVER

CONFIG-OPENVPN-SERVER

CONFIG-PPTP-SERVER

CONFIG-WIREGUARD-SERVER

CONFIG-WIREGUARD-PEER

Пример

```
rtt(config-if-gi)# security-zone trusted
```

security zone-pair

Данная команда используется для создания группы правил для пары зон безопасности.

Использование отрицательной формы команды (no) удаляет указанную группу правил.

Синтаксис

```
[no] security zone-pair <SOURCE-ZONE> <DESTINATION-ZONE> [ vrf <VRF> ]
```

Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик. Допустимые варианты зон:

- Пользовательская - созданная пользователем;
- any - служебная зона, характеризующая любые зоны;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик, созданная пользователем. Допустимые варианты зон:

- Пользовательская - созданная пользователем;
- any - служебная зона, описывающая любые зоны;
- self - служебная зона, описывающая трафик, предназначенный самому маршрутизатору (трафик не является транзитным);

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. Данный параметр возможно использовать только для пар зон "any"-"any" и "any"-"self";

Если при удалении используется значение параметра «all», то будут удалены все конфигурируемые пары зон безопасности.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# security zone-pair trusted self
rtt(config)# security zone-pair any self vrf VRF
```

show ip firewall counters

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

Синтаксис

```
show ip firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE> [
<ORDER> ] ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip firewall counters
 Zone-pair                    Rule        Action           Pkts        Bytes       Description
 ---------------------------  ----------  ---------------  ----------  ----------  --------------------
 any/any                      default     deny             0           0           --
 trusted/self                 1           permit           0           0           From local to router
 trusted/trusted              1           permit           0           0           --
```

show ip firewall sessions

Данная команда используется для просмотра активных IP-сессий.

Синтаксис

```
show ip firewall sessions [ vrf <VRF> ] [ protocol <TYPE> ] [ inside-source-address
<ADDR>] [ outside-source-address <ADDR> ] [ inside-destination-address <ADDR> ] [
outside-destination-address <ADDR> ] [ inside-source-port <PORT> ] [ outside-
source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port
<PORT> ] [ summary ] [ configuration ] [ expected ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IP-адреса источника приходящих пакетов;

inside-destination-address – команда для указания IP-адреса назначения приходящих пакетов;

outiside-source-address – команда для указания IP-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IP-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

summary – выводит суммарную статистику по IP-сессиям;

configuration – выводит настройку таймаутов и объема таблиц IP-сессий;

expected – команда для отображения сессий, ожидающих обработки других сессий.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip firewall sessions
  Codes: E - expected, U - unreplied,
         A - assured, C - confirmed

 Prot    Aging       Inside source          Inside destination     Outside source         Outside destination    Pkts        Bytes       Status
 -----   ----------  ---------------------  ---------------------  ---------------------  ---------------------  ----------  ----------  ------
 icmp    22          10.0.22.3              10.0.22.15             10.0.22.3              10.0.22.15             1           84          C
 udp     19          192.168.0.15:138       192.168.0.37:138       192.168.0.15:138       192.168.0.37:138       5           1100        UC
```

show ip firewall sessions tracking

Данной командой отображается настройка функционала отслеживания сессий уровня приложений.

Синтаксис

```
show ip firewall sessions tracking
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip firewall sessions tracking
Tracking Status:
    FTP:         Enabled
    H.323:       Enabled
    GRE:         Enabled
    PPTP:        Enabled
    NETBIOS-NS:  Enabled
    SIP:         Enabled
```

show ipv6 firewall counters

Данная команда используется для просмотра статистики по пакетам, проходящим между зонами, для которых не установлена сессия.

Синтаксис

```
show ipv6 firewall counters [ vrf <VRF> ] [ <SOURCE-ZONE> [ <DESTINATION-ZONE> [
<ORDER> ] ] ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены счетчики правил в указанном VRF;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 firewall counters trusted self
 Zone-pair                     Rule         Action           Pkts         Bytes        Description
 ----------------------------  -----------  ---------------  -----------  -----------  --------------------
 any/any                       default      deny             0            0            --
 trusted/self                  1            permit           0            0            From local to router
 trusted/trusted               1            permit           0            0            --
```

show ipv6 firewall sessions

Данная команда используется для просмотра активных IPv6-сессий.

Синтаксис

```
show ipv6 firewall sessions [ vrf <VRF> ] [summary] [ protocol <TYPE> ] [ inside-source-address <IPV6-ADDR>] [ outiside-source-address <IPV6-ADDR> ] [ inside-destination-address <IPV6-ADDR> ] [ outside-destination-address <IPV6-ADDR> ] [ inside-source-port <PORT> ] [ outside-source-port <PORT> ] [ inside-destination-port <PORT> ] [ outside-destination-port <PORT> ] [ expected ] [ summary ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра будут отображены активные сессии в указанном VRF;

<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<PORT> – TCP/UDP-порт, принимает значения [1..65535];

inside-source-address – команда для указания IPv6-адреса источника приходящих пакетов;

inside-destination-address – команда для указания IPv6-адреса назначения приходящих пакетов;

outiside-source-address – команда для указания IPv6-адреса источника отправляемых пакетов;

outside-destination-address – команда для указания IPv6-адреса назначения отправляемых пакетов;

inside-source-port – ключ для указания TCP/UDP-порта отправителя в приходящих пакетах;

outside-source-port – ключ для указания TCP/UDP-порта отправителя в отправляемых пакетах;

inside-destination-port – ключ для указания TCP/UDP-порта назначения в приходящих пакетах;

outside-destination-port – ключ для указания TCP/UDP-порта назначения в отправляемых пакетах;

expected – команда для отображения сессий, ожидающих обработки других сессий;

summary – выводит суммарную статистику по IPv6-сессиям.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 firewall sessions
  Codes: E - expected, U - unreplied,
         A - assured, C - confirmed

 Prot    Aging       Inside source          Inside destination     Outside source         Outside destination    Pkts        Bytes       Status
 -----   ----------  ---------------------  ---------------------  ---------------------  ---------------------  ----------  ----------  ------
 icmp6   13          fc00::2                fc00::1                fc00::2                fc00::1                --          --          C
 tcp     112         [fc00::2]:42156        [fc00::1]:22           [fc00::2]:42156        [fc00::1]:22           --          --          AC
```

show security zone

Данная команда используется для просмотра интерфейсов, входящих в зону безопасности.

Синтаксис

```
show security zone [<NAME>]
```

Параметры

<NAME> – имя зоны, задаётся строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show security zone
Zone name       Interfaces
-------------   ------------------------------------------
trusted         gi1/0/2-6, bridge 1, openvpn(open_test)
untrusted       gi1/0/1, te1/0/1-2, bridge 2, pptp(p)
```

show security zone-pair

Данная команда используется для просмотра списка пар зон.

Синтаксис

```
show security zone-pair
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show security zone-pair
From zone       To zone         VRF                          Description
-------------   -------------   --------------------------   ----------------------
trusted         untrusted       --                           Transit zone-pair
trusted         trusted         --                           --
trusted         self            WAN-2                        From WAN-2
untrusted       self            WAN-1                        From WAN-1
```

show security zone-pair configuration

Данная команда используется для просмотра правил для пары зон безопасности.

Синтаксис

```
show security zone-pair configuration <SOURCE-ZONE> <DESTINATION-ZONE> [<ORDER>]
```

Параметры

<SOURCE-ZONE> – зона безопасности, из которой поступает трафик;

<DESTINATION-ZONE> – зона безопасности, в которую поступает трафик;

<ORDER> – номер правила, принимает значения [1..10000]. При указании номера правила будет показана информация только по данному правилу.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show security zone-pair configuration trusted self
Order:                      1
Description:                --
Matching pattern:
    Protocol:               icmp
    Fragment:
    IP options:
    Source MAC:             any
    Destination MAC:        any
    Source address:         10.0.34.12
    Destination address:    10.0.34.90
```

```
    Destination NAT:          --
    Application:              --
Action:                       Permit
Status:                       Enabled
------------------------------------------------------------------------------
```

25.3. Управление логированием и защитой от сетевых атак

25.3.1. Управление защитой от сетевых атак

ip firewall screen dos-defense icmp-threshold

Данная команда включает защиту от ICMP flood атак. При включенной защите ограничивается количество icmp-пакетов всех типов в секунду для одного source адреса.

Использование отрицательной формы команды (no) отключает защиту от ICMP flood атак.

Синтаксис

```
ip firewall screen dos-defense icmp-threshold <NUM>
no ip firewall screen dos-defense icmp-threshold
```

Параметры

<NUM> – количество icmp-пакетов в секунду задается в диапазоне [1..10000].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense icmp-threshold 2000
```

ip firewall screen dos-defense land

Данная команда включает защиту от land-атак. При включенной защите блокируются пакеты с одинаковыми source и destination ip, и флагом SYN в заголовке TCP.

Использование отрицательной формы команды (no) отключает защиту от land-атак.

Синтаксис

```
[no] ip firewall screen dos-defense land
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense land
```

ip firewall screen dos-defense limit-session-destination

Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т. д.). Команда включает ограничение числа пакетов, передаваемых за секунду на один адреса назначения, которое смягчает DoS-атаки.

Использование отрицательной формы команды (no) снимает это ограничение.

Синтаксис

```
ip firewall screen dos-defense limit-session-destination <NUM>
no ip firewall screen dos-defense limit-session-destination
```

Параметры

<NUM> – ограничение количества ip-пакетов в секунду, задается в диапазоне [1..10000].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense limit-session-destination 1000
```

ip firewall screen dos-defense limit-session-source

Когда таблица IP-сессий хоста переполняется, он больше не в состоянии организовывать новые сессии и отбрасывает запросы (такое может происходить при различных DoS-атаках: SYN flood, UDP flood, ICMP flood, и т.д.). Команда включает ограничение числа пакетов, передаваемых за секунду с одного адреса источника, которое смягчает DoS-атаки.

Использование отрицательной формы команды (no) снимает это ограничение.

Синтаксис

```
ip firewall screen dos-defense limit-session-source <NUM>
no ip firewall screen dos-defense limit-session-source
```

Параметры

<NUM> – ограничение количества ip-пакетов в секунду задается в диапазоне [1..10000].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense limit-session-source 1000
```

ip firewall screen dos-defense syn-flood

Данная команда включает защиту от SYN flood атак. При включенной защите ограничивается количество TCP-пакетов с установленным флагом SYN в секунду для одного адреса назначения.

Использование отрицательной формы команды (no) отключает защиту от SYN flood атак.

Синтаксис

```
ip firewall screen dos-defense syn-flood <NUM> [src-dst]
```

```
no ip firewall screen dos-defense syn-flood
```

Параметры

<NUM> – максимальное количество TCP-пакетов с установленным флагом SYN в секунду задается в диапазоне [1..10000].

src-dst – ограничение количества TCP-пакетов с установленным флагом SYN на основании адреса источника и адреса назначения.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense syn-flood 100 src-dsr
```

ip firewall screen dos-defense udp-threshold

Данная команда включает защиту от UDP flood атак. При включенной защите ограничивается количество UDP-пакетов в секунду для одного адреса назначения.

Использование отрицательной формы команды (no) отключает защиту от UDP flood.

Синтаксис

```
ip firewall screen dos-defense udp-threshold <NUM>
no ip firewall screen dos-defense udp-threshold
```

Параметры

<NUM> – максимальное количество UDP-пакетов в секунду, задается в диапазоне [1..10000].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense udp-threshold
```

ip firewall screen dos-defense winnuke

Данная команда включает защиту от winnuke-атак. При включенной защите блокируются TCP-пакеты с установленным флагом URG и 139 портом назначения.

Использование отрицательной формы команды (no) отключает защиту от winnuke-атак.

Синтаксис

```
[no] ip firewall screen dos-defense winnuke
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen dos-defense winnuke
```

ip firewall screen spy-blocking fin-no-ack

Данная команда включает блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов с установленным флагом FIN и неустановленным флагом ACK.

Синтаксис

```
[no] ip firewall screen spy-blocking fin-no-ack
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking fin-no-ack
```

ip firewall screen spy-blocking icmp-type destination-unreachable

Данная команда включает блокировку всех ICMP-пакетов 3 типа (destination-unreachable), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 3 типа.

Синтаксис

```
[no] ip firewall screen spy-blocking icmp-type destination-unreachable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking icmp-type destination-unreachable
```

ip firewall screen spy-blocking icmp-type echo-request

Данная команда включает блокировку всех ICMP пакетов 8 типа (echo-request), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 8 типа.

Синтаксис

```
[no] ip firewall screen spy-blocking icmp-type echo-request
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking icmp-type echo-request
```

ip firewall screen spy-blocking icmp-type reserved

Данная команда включает блокировку всех ICMP-пакетов 2 и 7 типов (reserved), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 2 и 7 типов.

Синтаксис

```
[no] ip firewall screen spy-blocking icmp-type reserved
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking icmp-type reserved
```

ip firewall screen spy-blocking icmp-type source-quench

Данная команда включает блокировку всех ICMP-пакетов 4 типа (source quench), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 4 типа.

Синтаксис

```
[no] ip firewall screen spy-blocking icmp-type source-quench
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking icmp-type source-quench
```

ip firewall screen spy-blocking icmp-type time-exceeded

Данная команда включает блокировку всех ICMP-пакетов 11 типа (time exceeded), включая пакеты, сгенерированные самим маршрутизатором.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов 11 типа.

Синтаксис

```
[no] ip firewall screen spy-blocking icmp-type time-exceeded
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking icmp-type time-exceeded
```

ip firewall screen spy-blocking ip-sweep

Данная команда включает защиту от IP-sweep атак. При включенной защите, если в течение заданного в параметрах интервала приходит более 10 ICMP-запросов от одного источника, первые 10 запросов пропускаются маршрутизатором, а 11 и последующие отбрасываются на оставшееся время интервала.

Использование отрицательной формы команды (no) отключает защиту от ip-sweep атак.

Синтаксис

```
ip firewall screen spy-blocking ip-sweep <THRESHOLD> [ <TIME> ]
no ip firewall screen spy-blocking ip-sweep
```

Параметры

<THRESHOLD> – количество пакетов ip sweep атаки в секунду, задается в диапазоне [1..10000].

<TIME> – время блокировки в миллисекундах [1..1000000].

Значение по умолчанию

Отключено.

Без указания времени блокировки при включении устанавливается значение 10000.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking ip-sweep 1000
```

ip firewall screen spy-blocking port-scan

Данная команда включает защиту от port scan атак. Если в течение секунды на один источник приходит более <THRESHOLD> TCP-пакетов с флагом SYN на разные TCP-порты или более <THRESHOLD> UDP-пакетов на разные UDP-порты, то такое поведение фиксируется как port scan атака, и все последующие пакеты такого рода от источника блокируются на второй заданный интервал времени (<TIME>).

Использование отрицательной формы команды (no) отключает защиту от port scan атак.

Синтаксис

```
ip firewall screen spy-blocking port-scan <THRESHOLD> [ <TIME> ]
no ip firewall screen spy-blocking port-scan
```

Параметры

<THRESHOLD> – количество пакетов port scan атаки в секунду, указывается в диапазоне [1..10000].

<TIME> – время блокировки атакующего узла в миллисекундах [1..1000000].

Значение по умолчанию

Отключено.

Без указания времени блокировки при включении, устанавливается значение 10000.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking port-scan 100 1000
```

ip firewall screen spy-blocking spoofing

Данная команда включает защиту от IP spoofing атак. При включенной защите маршрутизатор проверяет пакеты на соответствие адреса источника и записей в таблице маршрутизации, и в случае несоответствия пакет отбрасывается. Например, если пакет с адресом источника 10.0.0.1/24 приходит на интерфейс Gi1/0/1, а в таблице маршрутизации данная подсеть располагается за интерфейсом Gi1/0/2, то считается, что адрес источника был подменен.

Использование отрицательной формы команды (no) отключает защиту от ip spoofing атак.

Синтаксис

```
[no] ip firewall screen spy-blocking spoofing
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking spoofing
```

ip firewall screen spy-blocking spoofing exclude <object-group>

Данная команда исключает из защиты от IP-spoofing атак указанную Object Group. В Object Group помещается список из допустимых адресов, которые будут игнорироваться. Команда используется вместе с основной ip firewall screen spy-blocking spoofing, которая включает защиту, иначе она не будет иметь эффекта. В случае, если на маршрутизатор приходит spoofing от разрешённых подсетей (например, частый опрос устройств в сети), то пакеты пропускаются.

Использование отрицательной формы команды (no) отключает исключение группы из защиты от ip-spoofing атак.

Синтаксис

```
[no] ip firewall screen spy-blocking spoofing exclude <object-group>
```

Параметры

<object-group> – список разрешённых для spoofing подсетей в Object Group.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking spoofing exclude <object-group>
```

ip firewall screen spy-blocking syn-fin

Данная команда включает блокировку TCP-пакетов, с установленными флагами SYN и FIN.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, с установленными флагами SYN и FIN.

Синтаксис

```
[no] ip firewall screen spy-blocking syn-fin
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking syn-fin
```

ip firewall screen spy-blocking tcp-all-flags

Данная команда включает блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN, PSH, URG. Обеспечивается защита от атаки XMAS.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN,PSH,URG.

Синтаксис

```
[no] ip firewall screen spy-blocking tcp-all-flag
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking tcp-all-flag
```

ip firewall screen spy-blocking tcp-no-flag

Данная команда включает блокировку TCP-пакетов, с нулевым полем flags.

Использование отрицательной формы команды (no) отключает блокировку TCP-пакетов, с нулевым полем flags.

Синтаксис

```
[no] ip firewall screen spy-blocking tcp-no-flag
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen spy-blocking tcp-no-flag
```

ip firewall screen suspicious-packets icmp-fragment

Данная команда включает блокировку фрагментированных ICMP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных ICMP-пакетов.

Синтаксис

```
[no] ip firewall screen suspicious-packets icmp-fragment
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets icmp-fragment
```

ip firewall screen suspicious-packets ip-fragment

Данная команда включает блокировку фрагментированных IP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных пакетов.

Синтаксис

```
[no] ip firewall screen suspicious-packets ip-fragment
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets ip-fragment
```

ip firewall screen suspicious-packets large-icmp

Данная команда включает блокировку ICMP-пакетов длиной более 1024 байт.

Использование отрицательной формы команды (no) отключает блокировку ICMP-пакетов длиной более 1024 байт.

Синтаксис

```
[no] ip firewall screen suspicious-packets large-icmp
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets large-icmp
```

ip firewall screen suspicious-packets syn-fragment

Данная команда включает блокировку фрагментированных TCP-пакетов, с флагом SYN.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных TCP-пакетов, с флагом SYN.

Синтаксис

```
[no] ip firewall screen suspicious-packets syn-fragment
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets syn-fragment
```

ip firewall screen suspicious-packets udp-fragment

Данная команда включает блокировку фрагментированных UDP-пакетов.

Использование отрицательной формы команды (no) отключает блокировку фрагментированных UDP-пакетов.

Синтаксис

```
[no] ip firewall screen suspicious-packets udp-fragment
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets udp-fragment
```

ip firewall screen suspicious-packets unknown-protocols

Данная команда включает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

Использование отрицательной формы команды (no) отключает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

Синтаксис

```
[no] ip firewall screen suspicious-packets unknown-protocols
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall screen suspicious-packets unknown-protocols
```

25.3.2. Управление оповещением о сетевых атаках

ip firewall logging interval

Данной командой устанавливается частота оповещения (по SNMP, syslog и в CLI) об обнаруженных и отраженных сетевых атаках. При детектировании атаки сообщение логируется мгновенно, но следующие

оповещения о данной конкретной атаке будут логироваться через заданный интервал времени, если атака будет носить продолжительный характер.

Использование отрицательной формы команды (no) возвращает таймер к значению по умолчанию.

Синтаксис

```
ip firewall logging interval <NUM>
no ip firewall logging interval
```

Параметры

<NUM> – интервал времени в секундах [30 .. 2147483647].

Значение по умолчанию

30

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall logging interval 60
```

logging firewall screen dos-defense

Данной командой включается механизм обнаружения и логирования DoS-атак через CLI, syslog и по SNMP. В связке с включенной защитой от атак будет производиться оповещение об отраженных DoS-атаках.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженных и отраженных DoS-атак.

Синтаксис

```
[no] logging firewall screen dos-defense <ATACK_TYPE>
```

Параметры

<ATACK_TYPE> – тип DoS-атаки, принимает значения:

- icmp-threshold;
- land;
- limit-session-destination;

- limit-session-source;
- syn-flood;
- udp-threshold;
- winnuke.

Подробное описание DoS-атак приведено в разделе Управление защитой от сетевых атак.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging firewall screen dos-defense land
```

logging firewall screen spy-blocking

Данной командой включается механизм обнаружения и логирования шпионской активности через CLI, syslog и по SNMP. В связке с включенной защитой от шпионской активности будет производиться оповещение о блокированной шпионской активности.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженной и блокированной шпионской активности.

Синтаксис

```
[no] logging firewall screen spy-blocking { <ATACK_TYPE> | icmp-type <ICMP_TYPE> }
```

Параметры

<ATACK_TYPE> – тип шпионской активности, принимает значения:

- fin-no-ack;
- ip-sweep;
- port-scan;
- spoofing;
- syn-fin;
- tcp-all-flag;
- tcp-no-flag.

<ICMP_TYPE> – тип icmp, принимает значения:

- destination-unreachable;
- echo-request;
- reserved;
- source-quench;
- time-exceeded.

Подробное описание шпионских активностей приведено в разделе Управление защитой от сетевых атак.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging firewall screen spy-blocking icmp-type echo-request
```

logging firewall screen suspicious-packets

Данной командой включается механизм обнаружения нестандартных пакетов и логирования через CLI, syslog и по SNMP. В связке с включенной защитой от нестандартных пакетов будет производиться также оповещение о блокировке нестандартных пакетов.

Использование отрицательной формы команды (no) отключает механизм обнаружения и логирования обнаруженных и блокированных нестандартных пакетов.

Синтаксис

```
[no] logging firewall screen suspicious-packets <PACKET_TYPE>
```

Параметры

< PACKET_TYPE> – тип нестандартных пакетов, принимает значения:

- icmp-fragment;
- ip-fragment;
- large-icmp;
- syn-fragment;
- udp-fragment;
- unknown-protocols.

Подробное описание защиты от нестандартных пакетов приведено в разделе Управление защитой от сетевых атак.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging firewall screen suspicious-packets icmp-fragment
```

show ip firewall screens counters

Данная команда позволяет просматривать статистику по обнаруженным сетевым атакам.

Синтаксис

```
show ip firewall screens counters
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show ip firewall screens counters
DDoS:
    Destination limit screen:  --
    Source limit screen:       --
    ICMP threshold screen:     1
    UDP threshold screen:      --
    SYN flood screen:          0
    Land attack screen:        1
    Winnuke attack screen:     --
Suspicious packets:
    ICMP fragmented screen:    --
    UDP fragmented screen:     --
    Large ICMP screen:         4
    Fragmented SYN screen:     --
    Unknown protocol screen:   --
    Fragmented IP screen:      --
```

```
Spying:
    Port scanning screen:      --
    IP sweep secreen:          --
    SYN-FIN screen:            --
    TCP all flags screen:      --
    FIN no ACK screen:         --
    TCP no flags screen:       --
    Spoofing screen:           --
ICMP types:
    ICMP reserved screen:      --
    ICMP quench screen:        --
    ICMP echo request screen:  --
    ICMP time exceeded screen: --
    ICMP unreachable screen:   --
```

25.4. Управление фильтрацией

default action

Данной командой устанавливается действие для обработки трафика неавторизованных пользователей.

Использование отрицательной формы команды (no) удаляет действие для обработки трафика неавторизованных пользователей.

Синтаксис

```
default action <ACTION> { redirect-url <REDIRECT-URL>}
no default action
```

Параметры

<ACTION> – действие, которое будет происходить для трафика, никак не описанного в профиле. Возможные значения:

- permit – пропускать http-запросы;
- deny – блокировать http-запросы;
- redirect – перенаправлять запросы на определенный URL-адрес.

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

Значение по умолчанию

Не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROXY-PROFILE

Пример

```
rtt(config-profile)# default action deny
```

description

Данной командой задаётся описание профиля фильтрации.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание, задаётся стройкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-PROXY-PROFILE

Пример

```
rtt(config-profile)# description "deny any"
```

filter

Данной командой включается режим фильтрации данных.

Использование отрицательной формы команды (no) режим фильтрации данных отключается.

Синтаксис

```
[no] filter <DATA-TYPE>
```

Параметры

<DATA-TYPE> – тип данных, подлежащих фильтрации. Может принимать значения:

- activex – фильтрация приложений с ActiveX;
- cookie – запретить веб-сайтам размещать cookie на пользовательских компьютерах;
- js – фильтрация всех страниц или приложений на основе Javascript.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROXY-PROFILE

Пример

```
rtt(config-profile)# filter cookie
```

log enable

Данной командой включается ведение журнала профиля прокси-сервера.

Использование отрицательной формы команды (no) ведение журнала отключается.

Синтаксис

```
[no] log enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROXY-PROFILE

Пример

```
rtt(config-proxy-profile)# log enable
```

ip http profile

Данной командой создается профиль фильтрации HTTP-трафика и производится переход в режим конфигурирования созданного профиля.

Использование отрицательной формы команды (no) удаляет профиль фильтрации HTTP-трафика.

Синтаксис

```
[no] ip http profile <NAME>
```

Параметры

<NAME> – имя профиля фильтрации HTTP-трафика, задается строкой длиной [1..31] символ.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip http profile HTTP_BLOCKED
```

ip http proxy

Данной командой на интерфейс или туннель назначается профиль фильтрации http-запросов.

Использование отрицательной формы команды (no) профиль фильтрации http-запросов удаляется.

Синтаксис

```
ip http proxy <NAME>
no ip http proxy
```

Параметры

<NAME> – имя профиля, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-LT

CONFIG-PPPOE

Пример

```
rtt(config-cellular-modem)# ip http proxy HTTP-BLOCKED
```

ip http proxy aaa das-profile

Данная команда используется для выбора профиля серверов динамической авторизации (DAS), на которые будут приходить CoA-запросы от PCRF об изменении списков URL.

Использование отрицательной формы команды (no) удаляет профиль динамической авторизации.

Синтаксис

```
ip http proxy aaa das-profile <NAME>
no ip http proxy aaa das-profile
```

Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

Значение по умолчанию

Профиль не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip http proxy aaa das-profile my_server
```

ip http proxy listen-ports

Данной командой устанавливается список tcp-портов для фильтрации http-запросов (вместо стандартных 80 и 8080).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip http proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

Параметры

<SERVICE_OBJ_GROUP_NAME> – профиль портов (object-group service). Задается строкой до 31 символа.

Значение по умолчанию

80, 8080

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip http proxy listen-ports HTTP-ADD
```

ip http proxy redirect-port

Данной командой определяется базовый порт HTTP Proxy-сервера на маршрутизаторе. Количество используемых портов соответствует числу ядер процессора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip http proxy redirect-port <PORT>
no ip http proxy redirect-port
```

Параметры

<PORT> – номер порта, указывается в диапазоне [1..65535].

Значение по умолчанию

3128

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip proxy http redirect-port 3001
```

ip http proxy server-url

Данной командой задается URL-адрес сервера, с которого будет загружаться список фильтрации.

Использование отрицательной формы команды (no) удаляет URL-адрес сервера.

Синтаксис

```
[no] ip http proxy server-url <URL>
```

Параметры

<URL> – URL-адрес ресурса, откуда необходимо скачивать файл со списком фильтрации. Задается строкой длиной [8..255] символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip http proxy server-url http://domain.example
```

ip https proxy

Данной командой на интерфейс или туннель назначается профиль фильтрации https-запросов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip https proxy <NAME>
no ip https proxy
```

Параметры

<NAME> – имя профиля. Задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-CELLULAR-MODEM

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-VTI

CONFIG-LT

CONFIG-PPPOE

Пример

```
rtt(config-cellular-modem)# ip https proxy HTTPS-BLOCKED
```

ip https proxy listen-ports

Данной командой устанавливается список дополнительных tcp-портов для фильтрации https-запросов (в дополнение к стандартному 433).

Использование отрицательной формы команды (no) удаляет ранее созданный список.

Синтаксис

```
[no] ip https proxy listen-ports <SERVICE_OBJ_GROUP_NAME>
```

Параметры

<SERVICE_OBJ_GROUP_NAME> – профиль портов (object-group service). Задается строкой до 31 символа.

Значение по умолчанию

443

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip https proxy listen-ports HTTPS-ADD
```

ip https proxy redirect-port

Данной командой определяется базовый порт HTTPS Proxy-сервера на маршрутизаторе. Количество используемых портов соответствует числу ядер процессора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip https proxy redirect-port <PORT>
no ip https proxy redirect-port
```

Параметры

<PORT> – номер порта, указывается в диапазоне [1..65535].

Значение по умолчанию

3129

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip proxy https redirect-port 3001
```

urls local

Данной командой в профиль фильтрации добавляется локальный список URL-адресов и действий для данного списка.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
urls local <URL_OBJ_GROUP_NAME> action { deny | permit | redirect redirect-url
<REDIRECT-URL> }
no urls local
```

Параметры

<URL_OBJ_GROUP_NAME> – список URL, для которых будет применяться действие;

permit – пропускать http-запросы на адреса, описанные в указанном списке URL;

deny – блокировать http-запросы на адреса, описанные в указанном списке URL;

redirect – перенаправлять запросы на адреса, описанные в указанном списке URL на определенный URL-адрес;

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROFILE

Пример

```
rtt(config-profile)# urls local OGU_DENY action deny
```

urls remote

Данной командой в профиль фильтрации добавляется ссылка на удаленный список URL-адресов и действий для данного списка.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
urls remote <FILE-NAME> action { deny | permit | redirect redirect-url <REDIRECT-URL> }
no urls remote
```

Параметры

<FILE-NAME> – имя файла, содержащее список URL и расположенного на сервере, прописанном в команде ip http proxy server-url (см. раздел ip http proxy server-url). Имя файла задается строкой до 31 символов;

permit – пропускать http-запросы на адреса, описанные в файле с указанным именем;

deny – блокировать http-запросы на адреса, описанные в файле с указанным именем;

redirect – перенаправлять запросы на адреса, описанные в файле с указанным именем на определенный URL-адрес.

<REDIRECT-URL> – URL-адрес для перенаправления http-запросов. Указывается только в режиме redirect.

Необходимый уровень привилегий

15

Командный режим

CONFIG-PROFILE

Пример

```
rtt(config-profile)# urls remote http-deny action deny
```

25.5. Управление системой предотвращения вторжений (IPS/IDS)

Данный функционал активируется только при наличии лицензии.

25.5.1. Общие команды IPS/IDS

clear content-filter cache

Данной командой выполняется очистка кэшированных ответов от Kaspersky-lab.

Синтаксис

```
clear content-filter cache [ url <URL> ]
```

Параметры

<URL> – URL, который необходимо удалить из кэша, задаётся строкой от 4 до 255 символов.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear content-filter cache url Rusteletech.ru
```

clear security ips counters

Данной командой выполняется сброс счетчиков работы правил сервиса IPS/IDS.

Синтаксис

```
clear security ips counters
```

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# clear security ips counters
rtt# 15: 2023-09-11T12:21:00.000+07:00 %IPS-I-INFO: Counters have been cleared
completely !
```

description

Данной командой выполняется изменение описания.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPS-CATEGORY

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-IPS-POLICY

CONFIG-IPS-UPGRADE-USER-SERVER

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-ips-upgrade-user-server)# description "Etnetera aggressive IP blacklist"
```

enable

Данной командой активируется сервис IPS/IDS и его правила.

Использование отрицательной формы команды (no) деактивирует сервис IPS/IDS.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

IPS/IDS-сервис не активирован.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

CONFIG-IPS-CATEGORY-RULE

CONFIG-IPS-CATEGORY-RULE-ADVANCED

CONFIG-CONTENT-PROVIDER

CONFIG-IPS-UPGRADE-USER-SERVER

Пример

```
rtt(config-ips)# enable
```

show content-filter cache

Данной командой выполняется просмотр записей контентной фильтрации, расположенных в кэше.

Синтаксис

```
show content-filter cache [url <URL> | vendor <NAME>]
```

Параметры

url <URL> – просмотр записей, отфильтрованных по URL. <URL> – текстовое поле, содержащее URL-ссылку длиной от 4 до 255 символов.

vendor <NAME> – просмотр записей, отфильтрованных по поставщику категорий. <NAME> – название поставщика категорий контентной фильтрации. Принимает значения:

Kaspersky-Lab – в текущей версии ПО в качестве поставщика категорий контентной фильтрации может выступать только Лаборатория Касперского.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show content-filter cache
URI                              Vendor              Categories                  Time to live
                                                                                  (d,h:m:s)
```

```
---------------------------   ---------------   ------------------------   -----------
rutube.ru                     kaspersky-lab     downloadable-content,      06,23:28:32
                                                hobbies-recreation,
                                                media-content
ya.ru                         kaspersky-lab     it-services, searchers     06,23:29:04
vtb.ru                        kaspersky-lab     finance, human-life,       06,23:46:01
                                                online-banks, payments,
                                                transactions
sber.ru                       kaspersky-lab     online-banks, payments     06,23:54:49
youtube.com                   kaspersky-lab     downloadable-content,      06,23:54:57
                                                media-content
```

show security ips content-provider

Данной командой выполняется просмотр информации об обновлении правил IPS/IDS, распространяемых по коммерческой лицензии.

Синтаксис

```
show security ips content-provider
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips content-provider
 Server: content-provider
                Last MD5 of received files:        93633ab9a73248ea50d58c25b1ac806c
                Next update: 2020-10-06 12:27:40
```

show security ips content-provider rules-info

Данной командой выполняется просмотр информации о категориях правил IPS/IDS, доступных по текущей коммерческой лицензии. При отсутствии действующей лицензии список будет пуст.

Синтаксис

```
show security ips content-provider rules-info
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips content-provider rules-info
Vendor : kaspersky
    Category : IoTURLsDF
        Count of rules : 8000
        Description : Kasperksy Lab IoTURLsDF feed
                      IoTURLsDF URL feed - a set of URLs with context covering malware that
infects IoT (Internet of Things) devices
    Category : MaliciousHashDF
        Count of rules : 1
        Description : Kasperksy Lab MaliciousHashDF feed
                      Malicious Hash feed - a set of hashes of malicious objects
    Category : PhishingURLsDF
        Count of rules : 11167
        Description : Kasperksy Lab PhishingURLsDF feed
                      Phishing URL feed - a set of URLs with context that cover phishing
websites and web pages
```

show security ips counters

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS.

Синтаксис

```
show security ips counters
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters
-----------------------------------------------
IPS general counters
-----------------------------------------------
Packets decoded by ips engine:               83971
Invalid packets decoded by ips engine:       0
Packets accepted by ips engine:              83977
Packets blocked by ips engine:               0
Packets replaced by ips engine:              0
Alerts generated:                            8
-----------------------------------------------
IPS Decoder engine
-----------------------------------------------
Packets decoded by ips engine:               83971
Bytes decoded by ips engine:                 125677543
Invalid packets decoded by ips engine:       0
IPv4 packets decoded by ips engine:          83971
IPv6 packets decoded by ips engine:          0
TCP packets decoded by ips engine:           75
UDP packets decoded by ips engine:           83891
SCTP packets decoded by ips engine:          0
ICMPv4 packets decoded by ips engine:        5
```

```
ICMPv6 packets decoded by ips engine:         0
PPP packets decoded by ips engine:            0
PPPoE packets decoded by ips engine:          0
GRE packets decoded by ips engine:            0
Teredo packets decoded by ips engine:         0
Average packets size decoded by ips engine:  1496
Maximum packets size decoded by ips engine:  1500
-----------------------------------------------
IPS Application Layer
-----------------------------------------------
HTTP Flow decoded by ips engine:              0
FTP Flow decoded by ips engine:               0
FTP-DATA Flow decoded by ips engine:          0
SMTP Flow decoded by ips engine:              0
TLS Flow decoded by ips engine:               0
SSH Flow decoded by ips engine:               0
IMAP Flow decoded by ips engine:              0
SMB Flow decoded by ips engine:               0
DCE/RPC flow over TCP decoded by ips engine: 0
DCE/RPC flow over UDP decoded by ips engine: 0
DNS flow over TCP decoded by ips engine:      0
DNS flow over UDP decoded by ips engine:      0
ENIP flow over TCP decoded by ips engine:     0
ENIP flow over UDP decoded by ips engine:     0
-----------------------------------------------
IPS Flow engine
-----------------------------------------------
TCP Flow decoded by ips engine:               1
UDP Flow decoded by ips engine:               1
ICMPv4 Flow decoded by ips engine:            1
ICMPv6 Flow decoded by ips engine:            0
Failed TCP Flow decoded by ips engine:        0
Failed UDP Flow decoded by ips engine:        1
-----------------------------------------------
IPS TCP engine
-----------------------------------------------
TCP sessions decoded by ips engine:           1
TCP SYN packets decoded by ips engine:        1
TCP SYN-ACK packets decoded by ips engine:    0
TCP RST packets decoded by ips engine:        0
TCP packets with invalid checksum:            0
TCP packets with wrong thread:                0
Packets with TCP header length too small:     0
TCP packets with invalid options:             0
```

show security ips counters application-layer

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS на уровне приложений.

Синтаксис

```
show security ips counters application-layer
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters application-layer
-----------------------------------------------
IPS Application Layer
-----------------------------------------------
HTTP Flow decoded by ips engine:              0
FTP Flow decoded by ips engine:               0
FTP-DATA Flow decoded by ips engine:          0
SMTP Flow decoded by ips engine:              0
TLS Flow decoded by ips engine:               0
SSH Flow decoded by ips engine:               0
IMAP Flow decoded by ips engine:              0
SMB Flow decoded by ips engine:               0
DCE/RPC flow over TCP decoded by ips engine: 0
DCE/RPC flow over UDP decoded by ips engine: 0
DNS flow over TCP decoded by ips engine:      0
DNS flow over UDP decoded by ips engine:      0
ENIP flow over TCP decoded by ips engine:     0
ENIP flow over UDP decoded by ips engine:     0
```

show security ips counters decoder

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS декодера.

Синтаксис

```
show security ips counters decoder
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters decoder
-----------------------------------------------
IPS Decoder engine
-----------------------------------------------
Packets decoded by ips engine:               83971
Bytes decoded by ips engine:                 125677543
Invalid packets decoded by ips engine:       0
IPv4 packets decoded by ips engine:          83971
IPv6 packets decoded by ips engine:          0
TCP packets decoded by ips engine:           75
UDP packets decoded by ips engine:           83891
SCTP packets decoded by ips engine:          0
ICMPv4 packets decoded by ips engine:        5
ICMPv6 packets decoded by ips engine:        0
PPP packets decoded by ips engine:           0
PPPoE packets decoded by ips engine:         0
GRE packets decoded by ips engine:           0
```

```
Teredo packets decoded by ips engine:        0
Average packets size decoded by ips engine:  1496
Maximum packets size decoded by ips engine:  1500
```

show security ips counters flow

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS на потоках.

Синтаксис

```
show security ips counters flow
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters flow
-----------------------------------------------
IPS Flow engine
-----------------------------------------------
TCP Flow decoded by ips engine:              1
UDP Flow decoded by ips engine:              1
ICMPv4 Flow decoded by ips engine:           1
ICMPv6 Flow decoded by ips engine:           0
Failed TCP Flow decoded by ips engine:       0
Failed UDP Flow decoded by ips engine:       1
-----------------------------------------------
```

show security ips counters general

Данной командой выполняется просмотр глобальных счетчиков работы правил сервиса IPS/IDS.

Синтаксис

```
show security ips counters general
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters general
```

```
-------------------------------------------
IPS general counters
-------------------------------------------
Packets decoded by ips engine:               83971
Invalid packets decoded by ips engine:       0
Packets accepted by ips engine:              83977
Packets blocked by ips engine:               0
Packets replaced by ips engine:              0
Alerts generated:                            8
-------------------------------------------
```

show security ips counters tcp

Данной командой выполняется просмотр счетчиков работы правил сервиса IPS/IDS для TCP-сессий.

Синтаксис

```
show security ips counters tcp
```

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips counters tcp
-------------------------------------------
IPS TCP engine
-------------------------------------------
TCP sessions decoded by ips engine:          1
TCP SYN packets decoded by ips engine:       1
TCP SYN-ACK packets decoded by ips engine:   0
TCP RST packets decoded by ips engine:       0
TCP packets with invalid checksum:           0
TCP packets with wrong thread:               0
Packets with TCP header length too small:    0
TCP packets with invalid options:            0
```

show security ips status

Данной командой выполняется просмотр числа правил, используемых в текущий момент системой IPS/IDS.

Синтаксис

```
show security ips status [detailed]
```

Параметры

detailed – показывает расширенную информацию об используемых правилах.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show security ips status

Rule files processed:         3
Rules successfully loaded:    21724
Rules failed:                 0


rtt# show security ips status detailed
Rule files processed:         3
Rules successfully loaded:    21724
Rules failed:                 0
Rules processed:              21727
IP-only inspecting:           1
Payload inspecting:           3980
Application layer inspecting: 18951
Decoder event:                0
```

show security ips user-server

Данной командой выполняется просмотр информации об обновлении правил IPS/IDS с пользовательских серверов обновлений.

Синтаксис

```
show security ips user-server [<WORD>]
```

Параметры

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# sh security ips user-server
Server name                     Files MD5                         Next update
--------------------------------   --------------------------------   --------------------
content-provider                93633ab9a73248ea50d58c25b1ac806c   2020-10-06 12:27:40
TH                              919f51bdf44052bfc0953362aef11c0d   2020-10-06 12:36:40
```

```
Traffic-ID                      e5e2f6472a397227c0d96f5df430a207   2020-10-06 12:36:40
Aggressive                      cfc3547b50f3f9fec366ba5a1e51cd1f   2020-10-06 12:36:40
JA3-Fingerprint                 439aa6e57c66826b92337672937d505b   2020-10-05 16:51:40
C2-Botnet                       39e118bd3884b3dc1df4ca3a03c05df1   2020-10-05 16:51:40
SSL-BlackList                   1d9c969f25791b9ee8c8c0ab8449d849   2020-10-05 16:51:40
ET-Open                         d53d92248a1f7cdc040d669a76cf27bc   2020-10-06 12:36:40
```

update security ips content-provider rules

Данной командой инициируется принудительное обновление правил IPS/IDS, распространяемых по коммерческой лицензии.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

Синтаксис

```
update security ips content-provider rules
```

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# update security ips content-provider rules
```

update security ips content-provider rules-info

Данной командой инициируется принудительный запрос информации о категориях правил IPS/IDS, доступных по текущей коммерческой лицензии.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

Синтаксис

```
update security ips content-provider rules-info
```

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# update security ips content-provider rules-info
```

update security ips user-server rules

Данной командой инициируется принудительное обновление правил IPS/IDS с пользовательского сервера обновлений.

Фактическое начало процедуры обновления правил происходит с некоторой задержкой после введения команды. Максимальная величина задержки составляет 5 минут.

Синтаксис

```
update security ips user-server rules <WORD>
```

Параметры

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# update security ips user-server rules ET-Open
```

25.5.2. Настройка политики IPS/IDS

category

Данной командой определяется категория правил IPS/IDS определённого вендора, распространяемых по коммерческой лицензии, и осуществляется переход в режим настройки данной категории.

Использование отрицательной формы команды (no) удаляет сконфигурированную категорию из настроек сервиса IPS/IDS.

Синтаксис

```
category <CATEGORY>
no category { <CATEGORY> | all }
```

Параметры

<CATEGORY> – категория правил.

Список доступных категорий можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-VENDOR

Пример

```
rtt(config-ips-policy-vendor)# category MobileBotnetCAndCDF
```

external network-group

Данной командой устанавливается профиль IP-адресов, которые сервис IPS/IDS будет считать ненадежными.

Сам профиль IP-адресов должен быть предварительно создан.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль из настроек сервиса IPS/IDS.

Синтаксис

```
external network-group <OBJ-GROUP-NETWORK-NAME>
no external network-group
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-POLICY

Пример

```
rtt(config-ips-policy)# external network-group WAN
```

protect network-group

Данной командой устанавливается профиль IP-адресов, которые будет защищать сервис IPS/IDS.

Сам профиль IP-адресов должен быть предварительно создан.

Использование отрицательной формы команды (no) удаляет сконфигурированный профиль из настроек сервиса IPS/IDS.

Синтаксис

```
protect network-group <OBJ-GROUP-NETWORK-NAME>
no protect network-group
```

Параметры

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-POLICY

Пример

```
rtt(config-ips-policy)# protect network-group LAN
```

rules

Данной командой указывается количество правил, которое необходимо скачать для данной категории.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Команда применима только для правил, распространяемых по коммерческой лицензии.

Синтаксис

```
rules { all | count <COUNT> | percent <PERCENT> | recomended }
no rules all
```

Параметры:

- all – данной командой указывается, что система IPS/IDS будет работать с полным набором правил данной категории;
- count <COUNT> – данной командой указывается действующее число правил данной категории, с которым будет работать система IPS/IDS:
    - <COUNT> – число правил. Минимальное значение 1, максимальное значение зависит от категории правил.

- percent <PERCENT> – данной командой указывается процентное соотношение от общего числа правил данной категории, с которым будет работать система IPS/IDS:
  - <PERCENT> – процент от общего числа правил.
- recomended – данной командой указывается, что система IPS/IDS будет использовать рекомендованное количество правил в данной категории. Рекомендованное количество составляет 42% от общего числа правил.

Максимальное число правил по категориям можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-VENDOR-CATEGORY

Пример

```
rtt(config-ips-vendor-category)# rules percent 25
```

rules action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего правилам данной категории.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Команда применима только для правил, распространяемых по коммерческой лицензии.

Синтаксис

```
rules action { alert | reject | pass | drop }
no rules action
```

Параметры:

- alert – прохождение трафика разрешается, и сервис IPS/IDS генерирует сообщение;
- reject – прохождение трафика запрещается. Если это TCP-трафик, отправителю и получателю посылается пакет TCP-RESET, для остального типа трафика посылается пакет ICMP-ERROR. Сервис IPS/IDS генерирует сообщение;
- pass – прохождение трафика разрешается;
- drop – прохождение трафика запрещается, и сервис IPS/IDS генерирует сообщение.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-VENDOR-CATEGORY

Пример

```
rtt(config-ips-vendor-category)# rules action drop
```

security ips policy

Данной командой создается политика настроек сервиса IPS/IDS с определенным именем и осуществляется переход в режим конфигурирования политики.

Использование отрицательной формы команды (no) удаляет сконфигурированную политику настроек сервиса IPS/IDS.

Синтаксис

```
[no] security ips policy <POLICY_NAME>
```

Параметры

<POLICY_NAME> – имя политики сервиса IPS/IDS, задаётся строкой до 31 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security ips policy OFFICE
```

vendor

Данной командой определяется вендор правил IPS/IDS, распространяемых по коммерческой лицензии, и осуществляется переход в режим настройки данного вендора.

Использование отрицательной формы команды (no) удаляет сконфигурированного вендора из настроек сервиса IPS/IDS.

Синтаксис

```
vendor <VENDOR>
no vendor <CATEGORY>
```

Параметры

<VENDOR> – вендор правил.

Список доступных вендоров можно посмотреть в контекстной подсказке или командой:

```
show security ips content-provider rules-info
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-POLICY

Пример

```
rtt(config-ips-policy)# vendor kaspersky
```

25.5.3. Настройка IPS

default-packet-size follow-mtu

Данной командой включается вычисление максимального MTU среди интерфейсов, участвующих в работе IPS. При включении опции значение default-packet-size становится равно вычисленному MTU + размер Ethernet-заголовка.

Использование отрицательной формы команды (no) сбрасывает значение MTU в конфигурации IPS/IDS на стандартное 1514 (размер MTU 1500 + заголовок ethernet 14).

Синтаксис

```
default-packet-size follow-mtu
[no] default-packet-size
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# default-packet-size follow-mtu
```

fail-close enable

При использовании данной команды передача трафика через интерфейс, на котором настроен сервис IPS/IDS, в момент начальной загрузки маршрутизатора, будет заблокирована до тех пор, пока сервис IPS/IDS не запустится и не применит хотя бы одно сконфигурированное или существующее правило.

Использование отрицательной формы команды (no) разрешает прохождение трафика при отсутствии загруженных правил IPS/IDS.

Синтаксис

```
[no] fail-close enable
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# fail-close enable
```

logging ips severity

Данной командой устанавливается уровень важности сообщения для логирования событий IPS/IDS. Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
logging ips severity <SEVERITY>
no logging ips severity
```

Параметры

<SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):

- emerg – в системе произошла критическая ошибка, система неработоспособна;
- alert – сигналы тревоги, необходимо немедленное вмешательство персонала;
- crit – критическое состояние системы, сообщение о событии;
- error – сообщения об ошибках;
- warning – предупреждения, неаварийные сообщения;
- notice – сообщения о важных системных событиях;
- info – информационные сообщения системы;
- debug – отладочные сообщения, предоставляют пользователю информацию для корректной настройки системы;
- none – отключает вывод syslog-сообщений.

Значение по умолчанию

info

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging ips severity error
```

logging remote-server

Данной командой задаются параметры удаленного сервера для отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch).

Использование отрицательной формы команды (no) останавливает отправку статистики.

Синтаксис

```
logging remote-server { <ADDR> | <IPV6-ADDR> } [ <TRANSPORT> ] [ <PORT> ] [ source-address { <SRC-ADDR> | <IPV6-SRC-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> } ]
no logging remote-server
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TRANSPORT> – протокол передачи данных, по умолчанию – UDP, принимает значения:

- TCP – передача данных осуществляется по протоколу TCP;
- UDP – передача данных осуществляется по протоколу UDP.

<PORT> – номер TCP/UDP-порта, опциональный параметр, принимает значения [1..65535], по умолчанию 514;

<SRC-ADDR> – IPv4-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv4-адрес интерфейса, с которого отправляются пакеты;

<IPV6-SRC-ADDR> – IPv6-адрес маршрутизатора, который будет использоваться в качестве IPv6-адреса источника в отправляемых syslog-пакетах, по умолчанию – IPv6-адрес интерфейса, с которого отправляются пакеты;

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# logging remote-server 192.168.0.101
```

logging update-interval

Данной командой устанавливается интервал отправки статистики работы сервиса IPS/IDS в формате EVE (elasticsearch).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
logging update-interval <INTERVAL>
no logging update-interval
```

Параметры

<INTERVAL> – интервал отправки статистики работы сервиса IPS/IDS, задаётся в минутах.

Значение по умолчанию

10 минут.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# logging update-interval 10
```

performance max

Данная команда позволяет сервису IPS/IDS использовать все ресурсы устройства для достижения максимальной производительности. Рекомендуется применять, когда устройство используется исключительно в качестве IPS/IDS. Не рекомендуется применять, когда помимо IPS/IDS устройство выполняет другие функции (маршрутизация, BRAS и т. д.).

Использование отрицательной формы команды (no) освобождает часть ресурсов устройства для использования другими сервисами.

Синтаксис

```
[no] performance max
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# perfomance max
```

policy

Данной командой назначается созданная ранее политика настроек сервиса IPS/IDS.

Использование отрицательной формы команды (no) снимает назначенную ранее политику настроек сервиса IPS/IDS.

Синтаксис

```
policy <POLICY_NAME>
no policy
```

Параметры

<POLICY_NAME> – имя политики сервиса IPS, задаётся строкой до 32 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# policy OFFICE
```

queue-limit

Данной командой определяется предельное количество пакетов для виртуальной очереди одного потока сервиса IPS/IDS. Больший размер очереди позволяет оптимизировать производительность при всплесках трафика, но слишком большая очередь может привести к снижению производительности.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
queue-limit <QUEUE-LIMIT>
no queue-limit
```

Параметры

<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [32..4096].

Значение по умолчанию

1024

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# queue-limit 2048
```

security ips

Команда используется для создания профиля сервиса IPS/IDS и перехода в режим его конфигурирования.

Синтаксис

```
security ips
```

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security ips
```

service-ips

Данная команда используется для включения сервиса IPS/IDS на сетевом интерфейсе.

Использование отрицательной формы команды (no) выключает сервис IPS/IDS на сетевом интерфейсе.

Синтаксис

```
service-ips { inline | monitor }
[no] service-ips
```

Параметры

inline – режим работы сервиса IPS/IDS, когда устройство ставится в разрыв сети. В этом режиме сервис IPS/IDS может детектировать и блокировать трафик;

monitor – режим работы сервиса IPS/IDS, когда устройство подключается к SPAN/RSPAN-порту. В этом режиме возможно только детектировать трафик, т. е. сервис работает в режиме IDS.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# service-ips inline
```

25.5.4. Настройка параметров кэширования записей контентной фильтрации

content-filter

Данной командой осуществляется переход в режим конфигурирования параметров кэширования записей контентной фильтрации.

Синтаксис

```
content-filter
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# content-filter
```

uri cache-size

Данная команда позволяет указать количество записей, которые будут храниться в памяти устройства.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] uri cache-size <NUMBER>
```

Параметры

<NUMBER> – значение размера кэша контент-фильтра в количестве записей. Принимает значение в диапазоне [1; 32768]. Значение по умолчанию – 32768.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CONTENT-FILTER

Пример

```
rtt(config-ips-content-filter)# uri cache-size 940
```

uri reachable-interval

Данная команда позволяет настроить среднее время, в течение которого запись URI всё ещё действительна.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] uri reachable-interval <DAYS>
```

Параметры

<DAYS> – количество дней, в течение которых запись URI всё ещё действительна. Принимает значения [1; 365]. Значение по умолчанию – 7.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CONTENT-FILTER

Пример

```
rtt(config-ips-content-filter)# uri reachable-interval 30
```

25.5.5. Настройка автообновления правил IPS/IDS, распространяемых по коммерческой лицензии

content-provider

Данной командой осуществляется переход в режим конфигурирования источника обновлений правил, распространяемых по коммерческой лицензии.

Синтаксис

```
content-provider
```

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# content-provider
```

host address

Данной командой задаётся адрес сервера обновлений правил, распространяемых по коммерческой лицензии.

Синтаксис

```
host address { <ADDR> | <IPV6-ADDR> | <WORD> }
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<WORD> – DNS-имя сервера, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# host address edm.rusteletech.ru
```

host port

Данной командой задаётся номер TCP-порта сервера обновлений правил, распространяемых по коммерческой лицензии.

Использование отрицательной формы команды (no) удаляет номер TCP-порта сервера обновлений правил, распространяемых по коммерческой лицензии.

Синтаксис

```
host port <PORT>
no host port
```

Параметры

<PORT> – номер TCP-порта, принимает значения [1..65535].

Значение по умолчанию

8098

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# host port 8098
```

location

Данной командой задаётся текстовое описание, которое передаётся на сервер EDM.

Использование отрицательной формы команды (no) удаляет это описание.

Синтаксис

```
location <WORD>
no location
```

Параметры

<WORD> – описание, задаётся строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# location "Server room in Moscow office"
```

reboot

Данной командой задаётся время перезагрузки устройства при получении системной лицензии. Устройство перезагружается при первом подключении к серверу обновлений правил, распространяемых по коммерческой лицензии.

При наличии функциональной лицензии IPS/IDS перезагрузки не происходит.

Синтаксис

```
reboot { immediately | time <TIME> }
```

Параметры

immediately – перезагружаться сразу после получения лицензии;

time <TIME> – перезагружаться в указанное время <TIME>;

<TIME> – время перезагрузки в формате HH:MM:SS.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# reboot time 05:00:00
```

storage-path

Данной командой задается путь на внешнем носителе, по которому будут сохраняться правила IPS/IDS в шифрованном виде, распространяемые по коммерческой лицензии.

Для использования с системой IPS/IDS на внешнем носителе должен быть создан раздел файловой системы в формате exFAT.

Использование отрицательной формы команды (no) останавливает сохранение правил.

Синтаксис

```
storage-path { usb://<USB-NAME>:/ | mmc://<MMC-NAME>:/ }
no storage-device
```

Параметры

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды show storage-devices usb;

<MMC-NAME> – имя подключенного MMC-носителя. Имя можно узнать в выводе команды show storage-devices mmc.

Необходимый уровень привилегий config-ips-upgrade-user-server

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# storage-device usb://DATA/IPS
```

system-name

Данной командой задаётся текстовое имя устройства, которое передаётся на сервер EDM.

Использование отрицательной формы команды (no) присваивает этому имени значение по умолчанию.

Синтаксис

```
system-name <WORD>
no system-name
```

Параметры

<WORD> – имя, задаётся строкой до 253 символа.

Значение по умолчанию

По умолчанию значение system-name совпадает с hostname.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# system-name main-office
```

upgrade interval

Команда задаёт частоту, с которой устройство будет проверять обновления правил IPS/IDS, распространяемых по коммерческой лицензии.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
upgrade interval <HOURS>
no upgrade interval
```

Параметры

<HOURS> – интервал обновлений в часах, от 1 до 240.

Значение по умолчанию

24

Необходимый уровень привилегий

15

Командный режим

CONFIG-CONTENT-PROVIDER

Пример

```
rtt(config-content-provider)# upgrade interval 36
```

25.5.6. Настройка автообновления правил IPS/IDS из внешних источников

auto-upgrade

Данной командой осуществляется переход в режим конфигурирования источников обновлений правил для сервиса.

Синтаксис

```
auto-upgrade
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# auto-upgrade
```

storage-path

Данной командой задается путь на внешнем носителе, по которому будут сохраняться правила IPS/IDS, загружаемые с пользовательских серверов обновлений.

Для использования с системой IPS/IDS на внешнем носителе должен быть создан раздел файловой системы в формате exFAT.

Использование отрицательной формы команды (no) останавливает сохранение правил.

Синтаксис

```
storage-path { usb://<USB-NAME>:/ | mmc://<MMC-NAME>:/ }
no storage-device
```

Параметры

<USB-NAME> – имя подключенного USB-носителя. Имя можно узнать в выводе команды show storage-devices usb;

<MMC-NAME> – имя подключенного MMC-носителя. Имя можно узнать в выводе команды show storage-devices mmc.

Необходимый уровень привилегий config-ips-upgrade-user-server

15

Командный режим

CONFIG-IPS

Пример

```
rtt(config-ips)# storage-device usb://DATA/
```

upgrade interval

Команда задаёт частоту, с которой устройство будет проверять обновления правил IPS/IDS и/или файла классификатора IPD/IDS для данного URL.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
upgrade interval <HOURS>
no upgrade interval
```

Параметры

<HOURS> – интервал обновлений в часах, от 1 до 240.

Значение по умолчанию

24

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-UPGRADE-USER-SERVER

Пример

```
rtt(config-ips-upgrade-user-server)# upgrade interval 36
```

url

Команда используется для задания URL-ссылки.

Использование отрицательной формы команды (no) удаляет ссылку из конфигурации источника обновлений правил IPS/IDS.

Синтаксис

```
url <URL>
no url
```

Параметры

<URL> – текстовое поле, содержащее URL-ссылку длиной от 8 до 255 символов.

В качестве URL-ссылки может быть указан:

- файл правил с расширение .rule,
- файл классификатора правил с именем classification.config,
- каталог на сервере, содержащий файлы правил и/или файл классификатора правил.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-UPGRADE-USER-SERVER

Пример

```
rtt(config-ips-upgrade-user-server)# url
https://sslbl.abuse.ch/blacklist/sslblacklist_tls_cert.rules
```

user-server

Данной командой задаётся имя пользовательского сервера обновлений правил IPS/IDS и осуществляется переход в режим конфигурирования параметров пользовательского сервера обновлений.

Использование отрицательной формы команды (no) удаляет пользовательский сервер обновлений правил IPS/IDS и все правила, полученные с этого сервера.

Синтаксис

```
user-server <WORD>
no user-server { <WORD> | all }
```

Параметры

<WORD> – имя сервера, задается строкой от 1 до 64 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-AUTO-UPGRADE

Пример

```
rtt(config-ips-auto-upgrade)# user-server SSL-BlackList
```

25.5.7. Настройка пользовательских правил IPS/IDS

action

Данная команда используется для указания действия, которое должно быть применено для трафика, удовлетворяющего заданным критериям.

Использование отрицательной формы команды (no) удаляет назначенное действие.

Синтаксис

```
action { alert | reject | pass | drop }
no action
```

Параметры

- alert – прохождение трафика разрешается, и сервис IPS/IDS генерирует сообщение;
- reject – прохождение трафика запрещается. Если это TCP-трафик, отправителю и получателю посылается пакет TCP-RESET, для остального типа трафика посылается пакет ICMP-ERROR. Сервис IPS/IDS генерирует сообщение;

- pass – прохождение трафика разрешается;
- drop – прохождение трафика запрещается, и сервис IPS/IDS генерирует сообщение.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# action reject
```

destination-address

Данной командой устанавливаются IP-адреса получателя, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
destination-address { ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME>
| policy-object-group { protect | external } | any }
no destination-address
```

Параметры

<ADDR> – IP-адрес получателя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];

<OBJ_GR_NAME> – имя профиля IP-адресов, который содержит IP-адреса назначения, задаётся строкой до 31 символа;

destination-address policy-object-group protect – устанавливает в качестве адресов назначения protect-адреса, определенные в политике IPS/IDS;

destination-address policy-object-group external – устанавливает в качестве адресов назначения external-адреса, определенные в политике IPS/IDS;

При указании значения «any» правило будет срабатывать для любого IP-адреса получателя.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# destination-address ip 10.10.10.1
```

destination-port

Данной командой устанавливается номер TCP/UDP-порта получателя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
destination-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no destination-port
```

Параметры

<PORT> – номер TCP/UDP-порта получателя, принимает значения [1..65535];

<OBJ_GR_NAME> – имя профиля TCP/UDP-портов получателя, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта получателя.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# destination-port 22
```

direction

Данной командой устанавливается направление потока трафика, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
direction { one-way | round-trip }
```

```
no direction
```

Параметры

- one-way – трафик передаётся в одну сторону;
- round-trip – трафик передаётся в обе стороны.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# direction one-way
```

ip dscp

Данной командой устанавливается значение кода DSCP, трафик которого будет обрабатываться в данном правиле.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip dscp <DSCP>
[no] ip dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip dscp 8
```

ip ftp command

Данной командой устанавливаются значения ключевых слов протокола FTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol ftp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip ftp command <COMMAND>
[no] ip ftp command
```

Параметры

<COMMAND> – может принимать следующие значения:

- <retr> – скачать файл;
- <stor> – залить файл;
- <mkd> – создать директорию;
- <rmd> – удалить директорию;
- <appe> – добавить в конец файла (с созданием);
- <dele> – удалить файл.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# protocol ftp
rtt(config-ips-category-rule)# ip ftp command allo
```

ip ftp-data command

Данной командой устанавливаются значения ключевых слов протокола FTP-DATA, для которых должно срабатывать правило.

Данная команда применима только для значения protocol ftp-data.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip ftp-data command <COMMAND>
[no] ip ftp-data command
```

Параметры

<COMMAND> – может принимать следующие значения:

- <retr> – скачать файл;
- <stor> – залить файл;
- <appe> – добавить в конец файла (с созданием).

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# protocol ftp-data
rtt(config-ips-category-rule)# ip ftp-data command stor
```

ip http

Данной командой устанавливаются значения ключевых слов протокола HTTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip http <COMMAND>
[no] ip http
```

Параметры

<COMMAND> – может принимать следующие значения:

- accept;
- accept-enc;
- accept-lang;
- client-body;
- connection;
- content-len;
- content-type;
- cookie;
- file-data;
- header;
- header-names;

- host;
- protocol;
- referer;
- request-line;
- response-line;
- server-body;
- start;
- stat-code;
- stat-msg;
- uri;
- urilen <VALUE>;
- urilen comparison-operator { greater-than | less-than};
- user-agent.

Значения и применение ключевых слов протокола HTTP подробно описаны в документации SNORT 2.X/Suricata 4.X.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload content "HTTP/1.0"
rtt(config-ips-category-rule)# ip http protocol
```

ip http content-filter

Данной командой назначается профиль категорий контентной фильтрации. Текущее правило будет срабатывать для http-сайтов, которые относятся к категориям заданным в этом профиле.

Сам профиль контентной фильтрации должен быть предварительно создан.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip http content-filter <NAME>
[no] ip http content-filter
```

Параметры

<NAME> – имя профиля контентной фильтрации, задаётся строкой до 31 символа.

any – правило будет срабатывать для http-сайтов любой категории.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip http content-filter Black-List
```

ip http method

Данной командой устанавливаются значения метода доступа по протоколу HTTP, для которых должно срабатывать правило.

Данная команда применима только для значения protocol http.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip http method <COMMAND>
[no] ip http method
```

Параметры

<COMMAND> – может принимать следующие значения:

- <GET> – запрашивает представление ресурса. Запросы с использованием этого метода могут только извлекать данные;
- <HEAD> – запрашивает ресурс так же, как и метод GET, но без тела ответа;
- <POST> – используется для отправки сущностей к определённому ресурсу;
- <PUT> – заменяет все текущие представления ресурса данными запроса;
- <DELETE> – удаляет указанный ресурс;
- <CONNECT> – устанавливает «туннель» к серверу, определённому по ресурсу;
- <OPTIONS> – используется для описания параметров соединения с ресурсом;
- <TRACE> – выполняет вызов возвращаемого тестового сообщения с ресурса;
- <PATCH> – используется для частичного изменения ресурса.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip http method get
```

ip icmp code

Данной командой устанавливается значение ICMP CODE, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip icmp code <CODE>
[no] ip icmp code
```

Параметры

<CODE> – значение CODE протокола ICMP, принимает значение в диапазоне [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp code 5
```

ip icmp code comparison-operator

Оператор сравнения для команды ip icmp code. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

Синтаксис

```
ip icmp code comparison-operator { greater-than | less-than }
[no] ip icmp code comparison-operator
```

Параметры

- greater-than – больше чем;
- less-than – меньше чем.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp code 5
rtt(config-ips-category-rule)# ip icmp code comparison-operator less-than
```

ip icmp id

Данной командой устанавливается значение ICMP ID, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip icmp id <ID>
[no] ip icmp id
```

Параметры

<ID> – значение ID протокола ICMP, принимает значение в диапазоне [0..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp id 65000
```

ip icmp sequence-id

Данной командой устанавливается значение ICMP sequence-ID, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip icmp sequence-id <SEQ-ID>
[no] ip icmp sequence-id
```

Параметры

<SEQ-ID> – значение Sequence-ID протокола ICMP, принимает значение в диапазоне [0..4294967295].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp sequence-id 8388608
```

ip icmp type

Данной командой устанавливается значение ICMP TYPE, при котором сработает правило.

Данная команда применима только для значения protocol icmp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip icmp type <TYPE>
[no] ip icmp type
```

Параметры

<TYPE> – значение TYPE протокола ICMP, принимает значение в диапазоне [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp type 12
```

ip icmp type comparison-operator

Оператор сравнения для команды ip icmp type. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

Синтаксис

```
ip icmp type comparison-operator { greater-than | less-than }
[no] ip icmp type comparison-operator
```

Параметры

- greater-than – больше чем;
- less-than – меньше чем.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip icmp type 14
rtt(config-ips-category-rule)# ip icmp code comparison-operator greater-than
```

ip protocol-id

Данной командой устанавливается идентификационный номер IP-протокола, трафик которого будет обрабатываться в данном правиле.

Данная команда применима только для значения protocol any.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip protocol-id <ID>
[no] ip protocol-id
```

Параметры

<ID> – идентификационный номер IP-протокола [1..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip protocol-id 250
```

ip tcp acknowledgment-number

Данной командой устанавливается значение TCP Acknowledgment-Number, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip tcp acknowledgment-number <ACK-NUM>
[no] ip tcp acknowledgment-number
```

Параметры

<ACK-NUM> – значение Acknowledgment-Number протокола TCP, принимает значение в диапазоне [0..4294967295].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip tcp acknowledgment-number 32
```

ip tcp sequence-id

Данной командой устанавливается значение TCP Sequence-ID, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip tcp sequence-id <SEQ-ID>
[no] ip tcp sequence-id
```

Параметры

<SEQ-ID> – значение Sequence-ID протокола TCP, принимает значение в диапазоне [0..4294967295].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip tcp sequence-id 2542
```

ip tcp window-size

Данной командой устанавливается значение TCP Window Size, при котором сработает правило.

Данная команда применима только для значения protocol tcp.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip tcp window-size <SIZE>
[no] ip tcp window-size
```

Параметры

<SIZE> – значение Window-Size протокола TCP, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip tcp window-size 50
```

ip tls content-filter

Данной командой назначается профиль категорий контентной фильтрации. Текущее правило будет срабатывать для https-сайтов, которые относятся к категориям заданным в этом профиле.

Сам профиль контентной фильтрации должен быть предварительно создан.

Данная команда применима только для значения protocol tls.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip tls content-filter <NAME>
[no] ip tls content-filter
```

Параметры

<NAME> – имя профиля контентной фильтрации, задаётся строкой до 31 символа.

any – правило будет срабатывать для http-сайтов любой категории.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip tls content-filter Black-List
```

ip ttl

Данной командой устанавливается значение времени жизни IP-пакета, трафик которого будет обрабатываться в данном правиле.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
ip ttl <TTL>
[no] ip ttl
```

Параметры

<TTL> – время жизни IP-пакета, принимает значения [1..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip ttl 8
```

ip ttl comparison-operator

Оператор сравнения для команды ip ttl. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

Синтаксис

```
ip ttl comparison-operator { greater-than | less-than }
[no] ip ttl comparison-operator
```

Параметры

- greater-than – больше чем;
- less-than – меньше чем.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# ip ttl 5
rtt(config-ips-category-rule)# ip ttl comparison-operator less-than
```

meta classification-type

Данная команда определяет классификацию события, которое сгенерирует сервис IPS/IDS, когда сработает правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
meta classification-type { not-suspicious | unknown | bad-unknown | attempted-recon
| successful-recon-limited | successful-recon-largescale | attempted-dos |
successful-dos | attempted-user | unsuccessful-user | successful-user | attempted-
admin | successful-admin | rpc-portmap-decode | shellcode-detect | string-detect |
suspicious-filename-detect | suspicious-login | system-call-detect | tcp-connection
| trojan-activity | unusual-client-port-connection | network-scan | denial-of-
service | non-standard-protocol | protocol-command-decode | web-application-
```

```
activity | web-application-attack | misc-activity | misc-attack | icmp-event |
inappropriate-content | policy-violation | default-login-attempt }
[no] meta classification-type
```

Параметры

- not-suspicious – неподозрительный трафик;
- unknown – неизвестный трафик;
- bad-unknown – потенциально плохой трафик;
- attempted-recon – попытка утечки информации;
- successful-recon-limited – утечка информации;
- successful-recon-largescale – масштабная утечка информации;
- attempted-dos – попытка отказа в обслуживании;
- successful-dos – отказ в обслуживании;
- attempted-user – попытка получения привилегий пользователя;
- unsuccessful-user – безуспешная попытка получения привилегий пользователя;
- successful-user – успешная попытка получения привилегий пользователя;
- attempted-admin – попытка получения привилегий администратора;
- successful-admin – успешная попытка получения привилегий администратора;
- rpc-portmap-decode – декодирование запроса RPC;
- shellcode-detect – обнаружен исполняемый код;
- string-detect – обнаружена подозрительная строка;
- suspicious-filename-detect – было обнаружено подозрительное имя файла;
- suspicious-login – была обнаружена попытка входа с использованием подозрительного имени пользователя;
- system-call-detect – обнаружен системный вызов;
- tcp-connection – обнаружено TCP-соединение;
- trojan-activity – был обнаружен сетевой троян;
- unusual-client-port-connection – клиент использовал необычный порт;
- network-scan – обнаружение сетевого сканирования;
- denial-of-service – обнаружение атаки отказа в обслуживании;
- non-standard-protocol – обнаружение нестандартного протокола или события;
- protocol-command-decode – обнаружена попытка шифрования;
- web-application-activity – доступ к потенциально уязвимому веб-приложению;
- web-application-attack – атака на веб-приложение;
- misc-activity – прочая активность;
- misc-attack – прочие атаки;
- icmp-event – общее событие ICMP;
- inappropriate-content – обнаружено неприемлемое содержание;
- policy-violation – потенциальное нарушение корпоративной конфиденциальности;
- default-login-attempt – попытка входа с помощью стандартного логина/пароля.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# meta classification-type misc-attack
```

meta log-message

Данная команда определяет текстовое сообщение, которое сгенерирует сервис IPS/IDS, когда сработает правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
meta log-message <MESSAGE>
[no] mera log-message
```

Параметры

<MESSAGE> – текстовое сообщение, задаётся строкой до 128 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# meta log-message "Possible SlowLorys attack"
```

payload content

Данной командой можно указать содержимое IP-пакетов, при совпадении с которым будет срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
payload content <CONTENT>
[no] payload content <CONTENT>
```

Параметры

<CONTENT> – текстовое сообщение, задаётся строкой до 1024 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload content "virus"
```

payload data-size

Данной командой устанавливается размер содержимого пакетов, при котором сработает правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
payload data-size <SIZE>
[no] payload data-size
```

Параметры

<SIZE> – размер содержимого пакетов, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload data-size 1024
```

payload data-size comparison-operator

Оператор сравнения для команды ip icmp type. Применим только совместно с этой командой.

Использование отрицательной формы команды (no) отменяет сравнение.

Синтаксис

```
payload data-size comparison-operator { greater-than | less-than }
[no] payload data-size comparison-operator
```

Параметры

- greater-than – больше чем;
- less-than – меньше чем.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload data-size 1024
rtt(config-ips-category-rule)# payload data-size comparison-operator less-than
```

payload depth

Данная команда указывает, сколько байтов с начала содержимого пакета будет проверять это правило. Команда используется только совместно с командой payload content. Может применяться совместно с командой payload offset.

Использование отрицательной формы команды (no) означает что будет проверяться все содержимое пакета на точное соответствие.

Синтаксис

```
payload depth <DEPTH>
[no] payload content depth
```

Параметры

<DEPTH> – число байт с начала содержимого пакета, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload content "abc"
rtt(config-ips-category-rule)# payload depth 3
```

Под действие правила попадут пакеты с содержимым «abcdef», «abc123», «abcabcabc» и т.д.

payload no-case

Данная команда указывает не различать прописные и заглавные буквы в описании содержимого пакетов. Команда используется только совместно с командой payload content.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
payload no-case
[no] payload content no-case
```

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload content "virus"
rtt(config-ips-category-rule)# payload no-case
```

Под действие правила попадут пакеты с содержимым «virus», «VIRUS», «ViRuS» и т.д.

payload offset

Данная команда указывает число байт смещения от начала содержимого пакета, с которого начнется проверка. Команда используется только совместно с командой payload content. Может применяться совместно с командой payload depth.

Использование отрицательной формы команды (no) означает что будет проверяться все содержимое пакета на точное соответствие.

Синтаксис

```
payload offset <OFFSET>
[no] payload content offset
```

Параметры

<OFFSET> – число байт смещения от начала содержимого пакета, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# payload content "abc"
rtt(config-ips-category-rule)# payload depth 6
rtt(config-ips-category-rule)# payload offset 3
```

Под действие правила попадут пакеты с содержимым «123abcdef», «defabc», «abcabcabc» и т. д.

protocol

Данной командой устанавливается имя IP-протокола, для которого должно срабатывать правило. Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
protocol { any | ip | icmp | http | tcp | tls | udp }
[no] protocol
```

Параметры

- any – правило будет срабатывать для любых протоколов;
- ip – правило сработает для протокола ip. В правиле можно настроить дополнительную фильтрацию командой ip protocol-id;
- icmp – правило сработает для протокола icmp. При выборе этого параметра значения source-port и destination-port должны быть any. В правиле можно настроить дополнительную фильтрацию командами ip icmp;
- http – правило сработает для протокола http. В правиле можно настроить дополнительную фильтрацию командами ip http;
- tcp – правило сработает для протокола tcp. В правиле можно настроить дополнительную фильтрацию командами ip tcp;
- tls – правило сработает для протокола https. В правиле можно настроить дополнительную фильтрацию командами ip tls;
- udp – правило сработает для протокола tcp. В правиле можно настроить дополнительную фильтрацию командами ip udp;
- ftp – правило сработает для протокола ftp. В правиле можно настроить дополнительную фильтрацию командами ip ftp;
- ftp-data – правило сработает для поля data протокола ftp. В правиле можно настроить дополнительную фильтрацию командами ip ftp-data.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# protocol udp
```

rule

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-IPS-CATEGORY-RULE. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..512].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY

Пример

```
rtt(config-ips-category)# rule 10
rtt(config-ips-category-rule)#
```

security ips-category user-defined

Данной командой создается набор пользовательских правил сервиса IPS/IDS с определенным именем и осуществляется переход в режим конфигурирования этого набора.

Использование отрицательной формы команды (no) удаляет сконфигурированную политику настроек сервиса IPS.

Синтаксис

```
[no] security ips-category user-defined <CATEGORY_NAME>
```

Параметры

<CATEGORY_NAME> – имя набора пользовательских правил сервиса IPS/IDS, задаётся строкой до 31 символов.

В текущей версии использование символа «_» (нижнее подчёркивание) в <CATEGORY_NAME> недопустимо.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# security ips-category user-defined PROTOCOL
rtt(config-ips-category)#
```

source-address

Данной командой устанавливаются IP-адреса отправителей, для которых должно срабатывать правило.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
source-address {ip <ADDR> | ip-prefix <ADDR/LEN> | object-group <OBJ_GR_NAME> |
policy-object-group { protect | external } | any }
no source-address
```

Параметры

<ADDR> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и LEN принимает значения [1..32];

<OBJ_GR_NAME> – имя профиля IP-адресов, который содержит IP-адреса отправителя, задаётся строкой до 31 символа;

destination-address policy-object-group protect – устанавливает в качестве адресов отправителя protect-адреса, определенные в политике IPS/IDS;

destination-address policy-object-group external – устанавливает в качестве адресов отправителя external-адреса, определенные в политике IPS/IDS.

При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# source-address ip-prefix 192.168.0.0/16
```

source-port

Данной командой устанавливается номер TCP/UDP-порта отправителя, для которого должно срабатывать правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
source-port { any | <PORT> | object-group <OBJ-GR-NAME> }
no source-port
```

Параметры

<PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535];

<OBJ_GR_NAME> – имя профиля TCP/UDP-портов отправителя, задаётся строкой до 31 символа.

При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# source-port 22
```

threshold count

Данной командой устанавливается пороговое значение количества пакетов, при котором сработает правило.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
threshold count <COUNT>
[no] threshold count
```

Параметры

<COUNT> – число пакетов, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# threshold count 1024
```

threshold second

Данной командой устанавливается интервал времени, для которого считается пороговое значение пакетов, при котором сработает правило. Команда используется только совместно с командой threshold count.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
threshold second <SECOND>
[no] threshold second
```

Параметры

<SECOND> – интервал времени в секундах, принимает значение в диапазоне [1..65535].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# threshold second 1
```

threshold track

Данной командой устанавливается по адресу отправителя или получателя будут считаться пакеты, для которых устанавливаются пороговые значения. Команда используется только совместно с командой threshold count.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
threshold track { by_src | by_dst }
[no] threshold track
```

Параметры

- by_src – считать пороговое значение для пакетов с одинаковым IP отправителя;
- by_dst – считать пороговое значение для пакетов с одинаковым IP получателя.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# threshold track by-src
```

threshold type

Данной командой устанавливается метод обработки пороговых значений. Команда используется только совместно с командой threshold count.

Использование отрицательной формы команды (no) удаляет назначение.

Синтаксис

```
threshold type { treshhold | limit | both }
[no] threshold type
```

Параметры

- threshold – выдавать сообщение каждый раз по достижении порога;
- limit – выдавать сообщение не чаще <COUNT> раз за интервал времени <SECOND>;
- both – комбинация threshold и limit. Сообщение будет генерироваться, если в течение интервала времени <SECOND> было <COUNT> или более пакетов, подходящих под условия правила, и сообщение будет отправлено только один раз в течение интервала времени.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE

Пример

```
rtt(config-ips-category-rule)# threshold count 1024
rtt(config-ips-category-rule)# threshold second 1
rtt(config-ips-category-rule)# threshold track by-src
rtt(config-ips-category-rule)# threshold type treshold
```

Сообщение будет генерироваться на каждый X*1025 пакет, приходящий за 1 секунду с одного IP-адреса.

25.5.8. Настройка расширенных пользовательских правил

rule-advanced

Данная команда используется для создания правила и перехода в режим конфигурирования CONFIG-IPS-CATEGORY-RULE-ADVANCED. Правила обрабатываются устройством в порядке возрастания их номеров.

Использование отрицательной формы команды (no) удаляет указанное правило.

Синтаксис

```
[no] rule-advanced <ORDER>
```

Параметры

<ORDER> – номер правила, принимает значения [1..4294967295].

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE-ADVANCED

Пример

```
rtt(config-ips-category)# rule-advanced 10
rtt(config-ips-category-rule-advanced)#
```

rule-text

Данная команда описывает правило обработки трафика в формате SNORT 2.X/Suricata 4.X.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
rule-text <LINE>
[no] rule-text
```

Параметры

<LINE> – текстовое сообщение в формате SNORT 2.X/Suricata 4.X, задаётся строкой до 1024 символов.

При написании правил в тексте правила необходимо использовать только двойные кавычки (символ "), а само правило необходимо заключать в одинарные кавычки (символ ').

Необходимый уровень привилегий

15

Командный режим

CONFIG-IPS-CATEGORY-RULE-ADVANCED

Пример

```
rtt(config-ips-category-rule-advanced)# rule-text 'alert tcp any any -> $HOME_NET any (msg:
"ATTACK [PTsecurity] Attempt to crash named using malformed RNDC packet"; flow: established,
to_server; content:"_auth"; depth: 20; fast_pattern; content: !"|02 00 00 00|"; within: 4;
content: "_ctrl"; content: "_ser"; content: "_tim"; content: "_exp"; reference: cve, 2016-
1285; classtype: attempted-dos; reference: url, github.com/ptresearch/AttackDetection;
metadata: Open Ptsecurity.com ruleset; sid: 10000005; rev: 3; )'
```

26. УПРАВЛЕНИЕ QOS

class

Данная команда используется для привязки указанного QoS-класса к политике и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет привязку класса к политике.

Синтаксис

```
[no] class <NAME>
```

Параметры

<NAME> – имя привязываемого класса, задается строкой до 31 символа. При указании значения «class-default» в данный класс попадает трафик неклассифицированный на входе.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

Пример

```
rtt(config-policy-map)# class telnet_traffic
rtt(config-class-policy-map)#
```

class-map

Данной командой создается класс QoS и осуществляется переход в режим настройки параметров класса.

Использование отрицательной формы команды (no) удаляет созданный класс.

Привила, описанные внутри класса, работают по принципу логического "ИЛИ".

Пример:

```
RTT(config)# class-map VOICE
RTT(config-class-map)# match dscp 12
RTT(config-class-map)# match dscp 56
RTT(config-class-map)# exit
```

Трафик, содержащий метки DSCP 12 или DSCP 56, будет попадать под критерий класса "VOICE".

Синтаксис

```
[no] class-map <NAME>
```

Параметры

<NAME> – имя создаваемого класса, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# class-map telnet_traffic
```

compression header ip tcp

Данной командой включается протокол компрессии tcp-заголовков для трафика отдельного класса.

Использование отрицательной формы команды (no) отменяет использование протокола компрессии tcp-заголовков для трафика отдельного класса.

Синтаксис

```
[no] compression header ip tcp
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# compression header ip tcp
```

description

Данной командой определяется описание классификатора QoS.

Использование отрицательной формы команды (no) удаляет описание.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание классификатора, задаётся строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-CLASS-MAP

CONFIG-POLICY-MAP

Пример

```
rtt(config-class-map)# description "VoIP"
```

fair-queue

Данной командой определяется предельное количество потоков для данного класса. Параметр характерен для очередей SFQ и FIFO.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
fair-queue <QUEUE-LIMIT>
no fair-queue
```

Параметры

<QUEUE-LIMIT> – предельное количество виртуальных очередей, принимает значения в диапазоне [16..4096].

Значение по умолчанию

16

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# fair-queue 200
```

ip firewall sessions classification enable

Командой выполняется включение классифицирования сессий на основе политики QoS.

Использование отрицательной формы команды (no) отключает классифицирование сессий.

Синтаксис

```
[no] ip firewall sessions classification enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip firewall sessions classification enable
```

match access-group

Команда используется для привязки списка контроля доступа (ACL), по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка контроля доступа к данному классу.

Синтаксис

```
[no] match access-group {<NAME> | all }
```

Параметры

<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

Пример

```
rtt(config-class-map)# match access-group acl-ssh-traffic
```

match application

Команда используется для привязки списка приложений, по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка приложений к данному классу.

Классификация QoS по спискам приложений работает, только если firewall настроен на работы в StateLess-режиме.

Синтаксис

```
[no] match application { <NAME> | all }
```

Параметры

<NAME> – имя списка приложений, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

Пример

```
rtt(config-class-map)# match application APP-PERMIT
```

match dscp

Данной командой устанавливается значение кода DSCP, трафик которого будет обрабатываться в данном class-map.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
[no] match dscp <DSCP>
```

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

При удалении возможно использование ключа "all" для удаления всех заданных dscp.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

Пример

```
rtt(config-class-map)# match dscp 55
```

match mpls experimental topmost

Данной командой устанавливается значение поля EXP в заголовке MPLS, трафик которого будет обрабатываться в данном class-map.

Использование отрицательной формы команды (no) отменяет назначение.

При использовании данной команды в class-map запрещена настройка других match и set команд.

Синтаксис

```
[no] match mpls experimental topmost <VALUE>
```

Параметры

<VALUE> – значение поля EXP, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

```
rtt(config-class-map)# match mpls experimental topmost 1
```

match protocol nhrp

Данной командой включается определение сообщений протокола NHRP в настраиваемый class-map.

Команда имеет практическую применимость только в том случае, когда class-map с данной командой используется в policy-map, назначенной на GRE интерфейс в режиме multipoint и включенной поддержкой протокола NHRP.

Использование отрицательной формы команды (no) отменяет назначение.

При использовании данной команды в class-map запрещена настройка других match и set команд.

Синтаксис

```
[no] match protocol nhrp
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

```
rtt(config-class-map)# match protocol nhrp
```

match url

Команда используется для привязки списка url, по которому будет определяться отношение входящего трафика к конфигурируемому классу.

Использование отрицательной формы команды (no) удаляет привязку списка url.

Классификация QoS по спискам url работает, только если firewall настроен на работы в StateLess-режиме.

Синтаксис

```
[no] match url { <NAME> | all }
```

Параметры

<NAME> – имя списка url, задаётся строкой до 31 символа.

При удалении возможно использование ключа "all" для удаления всех заданных списков доступа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

Пример

```
rtt(config-class-map)# match url URL-PERMIT
```

mode

Данной командой определяется режим работы класса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
mode <MODE>
no mode
```

Параметры

<MODE> – режим класса:

- fifo – режим FIFO (First In, First Out);
- gred – режим GRED (Generalized RED);
- red – режим RED (Random Early Detection);
- sfq – режим SFQ (очередь SFQ распределяет передачу пакетов на базе потоков).

Значение по умолчанию

FIFO

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# mode red
```

police

Данной командой настраивается прохождение, отбрасывание или маркировка пакетов, соответствующих конфигурируемому классу или политики в целом, исходя из заполненности токенами корзин conform и exceed.

Использование отрицательной формы команды (no) удаляет настроенный параметр.

Синтаксис

```
police <RATE> [burst-conforming <BURST-CONFORM>] [conform-action <CONFORM-ACTION>]
[exceed-action <EXCEED-ACTION>] [burst-excess <BURST-EXCEED> [violate-action
<VIOLATE-ACTION>]]
no police
```

Параметры

<RATE> – скорость пополнения токенами conform корзины, Кбит/с;

<BURST-CONFORM> – размер conform-корзины, Байт;

<BURST-EXCEED> – размер excess-корзины, Байт;

<CONFORM-ACTION> – действие, которое необходимо выполнить с пакетом, для которого имеются токены conform-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP> };

<EXCEED-ACTION> – действие, которое необходимо выполнить с пакетом, при исчерпании токенов conform-корзины, но имеются токены excess-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP>};

<VIOLATE-ACTION> – действие, которое необходимо выполнить с пакетом, для которого исчерпаны токены excess-корзины, принимает значения { permit | deny | set-cos <COS> | set-dscp <DSCP> };

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];

<DSCP> – значение кода DSCP, принимает значения [0..63].

Значение по умолчанию

<RATE> – отсутствует;

<BURST-CONFORM> – 1500;

<BURST-EXCEED> – отсутствует;

<CONFORM-ACTION> – permit;

<EXCEED-ACTION> – deny;

<VIOLATE-ACTION> – deny;

<COS> – отсутствует;

<DSCP> – отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-policy-map)# police 256
rtt(config-class-policy-map)# police 256 burst-excess 3000 exceed-action set-dscp
63
```

policy-map

Данной командой создается политика QoS и осуществляется переход в режим настройки параметров политики.

Использование отрицательной формы команды (no) удаляет созданную политику.

Синтаксис

```
[no] policy-map <NAME>
```

Параметры

<NAME> – имя создаваемой политики, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# policy-map input_policy
```

priority class

Данной командой задается приоритет класса в WRR-процессе. Классы с наибольшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
priority class <PRIORITY>
no priority
```

Параметры

<PRIORITY> – приоритет класса в WRR-процессе, принимает значения [1..8].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# priority class 5
```

priority level

Данной командой класс переводится в режим Strict Priority и задается приоритет класса. Классы с наибольшим приоритетом обрабатываются в первую очередь.

Использование отрицательной формы команды (no) переводит класс в режим WRR.

Синтаксис

```
priority level <PRIORITY>
no priority
```

Параметры

<PRIORITY> – приоритет класса в Strict Priority-процессе, принимает значения: [1..8].

Значение по умолчанию

Класс работает в режиме WRR, приоритет не задан.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# priority level 5
```

priority-queue out

Данная команда задаёт предельное количество пакетов в очереди.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
priority-queue out <QUEUE> limit <LIMIT>
no priority-queue out <QUEUE> limit
```

Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [100-1000].

Значение по умолчанию

500

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI[1]

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL (данный функционал поддерживается только на маршрутизаторах R100/200)

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-L2TPV3

Пример

```
rtt(config-if)# priority-queue out 1 limit 800
```

priority-queue out num-of-queues

Данная команда задает количество приоритетных очередей от восьмой к первой. Оставшиеся очереди являются взвешенными.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
priority-queue out num-of-queues <VALUE>
no priority-queue out num-of-queues
```

Параметры

<VALUE> – количество очередей, принимает значение [1..8], где:

- 0 – все очереди участвуют в WRR (WRR – механизм обработки очередей на основе веса);
- 8 – все очереди обслуживаются как «strict priority» (strict priority – приоритетная очередь обслуживается сразу, как только появляются пакеты).

Значение по умолчанию

8

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# priority-queue out num-of-queues 5
```

qos dscp-mutation

Данной командой включается применение изменений кодов DSCP в соответствии с таблицей DSCP-Mutation. Коды DSCP изменяются только для входящего трафика доверенных портов в режиме QoS Basic.

Использование отрицательной формы команды (no) отключает изменение кодов DSCP.

Синтаксис

```
[no] qos dscp-mutation
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos wrr-queue 3 bandwidth 130
```

qos enable

Данной командой включается сервис QoS на интерфейсе. Если к интерфейсу не привязана политика QoS (привязка политик описана в разделе service-policy), то интерфейс работает в режиме Basic QoS, иначе Policy-based QoS.

Basic QoS – классификация трафика выполняется на основе кодов DSCP и/или 802.1p в зависимости от выбранного режима доверия (команда описана в разделе qos trust). Трафик направляется в очереди в соответствии с таблицами DSCP-Queue и/или CoS-Queue.

Policy-based QoS – классификация и направление трафика в очереди выполняется на основе QoS-политик. В каждой политике определяется набор классов, на которые необходимо разделить трафик. Отношение трафика к определенному классу политики определяется на входе в маршрутизатор

правилами ACL (привязка ACL описана в разделе match access-group), для этого назначается QoS-политика на входящее направление. Для ограничения полосы ранее классифицированного трафика и других функций QoS-политика назначается на исходящее направление.

Использование отрицательной формы команды (no) выключает сервис QoS на интерфейсе.

Синтаксис

```
[no] qos enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-SUBTUNNEL

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

Пример

```
rtt(config-if-gi)# qos enable
```

qos map cos-queue

Данная команда устанавливает соответствие между значениями кодов 802.1p входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

Синтаксис

```
qos map cos-queue <COS> to <QUEUE>
no qos map dscp-queue <COS>
```

Параметры

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];

<QUEUE> – идентификатор очереди, принимает значения [1..8].

Значения по умолчанию:

CoS: (0), очередь 1

CoS: (1), очередь 2

CoS: (2), очередь 3

CoS: (3), очередь 4

CoS: (4), очередь 5

CoS: (5), очередь 6

CoS: (6), очередь 7

CoS: (7), очередь 8

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos map cos-queue 7 to 5
```

qos map dscp-mutation

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

Синтаксис

```
qos map dscp-mutation <DSCP> to <DSCP>
no qos map dscp-mutation <DSCP>
```

Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63].

Значения по умолчанию:

Значения кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства совпадают.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos map dscp-mutation 10 to 40
```

qos map dscp-queue

Данная команда устанавливает соответствие между значениями кодов DSCP входящих пакетов и исходящими очередями.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

Синтаксис

```
qos map dscp-queue <DSCP> to <QUEUE>
no qos map dscp-queue <DSCP>
```

Параметры

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];

<QUEUE> – внутренняя очередь, в которую необходимо отправлять пакеты с ранее описанным DSCP, принимает значение [1..8].

Значения по умолчанию:

DSCP: (0-7), очередь 1

DSCP: (8-15), очередь 2

DSCP: (16-23), очередь 3

DSCP: (24-31), очередь 4

DSCP: (32-39), очередь 5

DSCP: (40-47), очередь 6

DSCP: (48-55), очередь 7

DSCP: (56-63), очередь 8

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos map dscp-queue 42 to 5
```

qos queue default

Данная команда устанавливает номер очереди по умолчанию, в которую попадает весь трафик, кроме IP, в режиме доверия DSCP-приоритетам в случае с Basic QoS, а также неклассифицированный трафик в случае с Policy-based QoS.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
qos queue default <QUEUE>
no qos queue default
```

Параметры

<QUEUE> – идентификатор очереди, принимает значения [1..8].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos queue default 3
```

qos trust

Данной командой устанавливается режим доверия к значениям кодов 802.1p и DSCP во входящих пакетах для Basic QoS-режима работы интерфейса.

Использование отрицательной формы команды (no) устанавливает соответствие по умолчанию.

Синтаксис

```
qos trust <MODE>
no qos trust
```

Параметры

<MODE> – режим доверия к значениям кодов 802.1p и DSCP, принимает одно из следующих значений:

- dscp – режим доверия значениям кодов DSCP в IP-заголовке. Не IP-пакеты будут направлены в очередь по умолчанию (команда описана в разделе qos queue default);
- cos – режим доверия значениям кодов 802.1p в теге 802.1q. Нетегированные пакеты будут направлены в очередь по умолчанию (команда описана в разделе qos queue default);
- cos-dscp – режим доверия значениям кодов DSCP для IP-пакетов и значениям кодов 802.1p для остальных пакетов.

Значения по умолчанию

Режим доверия значениям кодов DSCP (dscp).

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos trust cos-dscp
```

qos wrr-queue

Данной командой определятся веса для соответствующих взвешенных очередей.

Использование отрицательной формы команды (no) устанавливает значение веса для указанной очереди по умолчанию.

Синтаксис

```
qos wrr-queue <QUEUE> bandwidth <WEIGHT>
no qos wrr-queue <QUEUE>
```

Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<WEIGHT> – значение веса, принимает значение [1..255].

Значение по умолчанию

1

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# qos wrr-queue 3 bandwidth 130
```

queue-limit

Данной командой определяется предельное количество пакетов для виртуальной очереди.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
queue-limit <QUEUE-LIMIT>
no queue-limit
```

Параметры

<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [2..4096].

Значение по умолчанию

127

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# queue-limit 200
```

random-detect

Данной командой определяются параметры RED.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
random-detect <LIMIT> <MIN> <MAX> <APS> <APS-NUM> <PROBABILITY>
no random-detect
```

Параметры

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000];

<APS-NUM> – количество пакетов среднего размера, разрешенных для кратковременного пропускания, принимает значение в диапазоне [0..10000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

<MAX> > 2 * <MIN> <LIMIT> >3 * <MAX>

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# random-detect 9000 1500 3000 1000 10 100
```

random-detect queue

Данной командой определяются параметры GRED.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
random-detect queue <QUEUE-NUM> [ dscp <DSCP> | precedence <IPP> ] <LIMIT> <MIN>
<MAX> <APS> <APS-NUM> <PROBABILITY>
no random-detect queue <PRECEDENCE>
```

Параметры

<QUEUE-NUM> – номер очереди [1..16];

<DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];

<IPP> – значение кода IP Precedence, принимает значения [0..7];

<PRECEDENCE> – значение IP Precendence [0..7];

<LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];

<APS> – средний размер пакета в байтах, принимает значение в диапазоне [1..10000000];

<APS-NUM> – количество пакетов среднего размера, разрешенных для кратковременного пропускания, принимает значение в диапазоне [0..10000000];

<PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].

При указании значений должны выполняться следующие правила:

<MAX> > 2 * <MIN>

<LIMIT> > 3 * <MAX>

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# random-detect queue 1 precedence 2 9000 1500 3000
1000 10 100
```

random-detect queue default

Данной командой определяется очередь по умолчанию для механизма GRED.

Для применения данной команды предварительно необходимо настроить команду:

```
random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX>
<APS> <APS-NUM> <PROBABILITY>
```

После применения команд:

```
random-detect queue <QUEUE-NUM> dscp <DSCP>/precedence <0-7> <LIMIT> <MIN> <MAX>
<APS> <APS-NUM> <PROBABILITY>

random-detect queue default <QUEUE-NUM>
```

поля dscp/precedence в заголовке IP-пакетов данной очереди будут игнорироваться.

Более подробно параметры, указываемые в команде `random-detect queue`, описаны в разделе random-detect queue.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
random-detect queue default <QUEUE-NUM>
no random-detect queue default <QUEUE-NUM>
```

Параметры

<QUEUE-NUM> – номер очереди [1..16].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# random-detect queue default
```

rate-limit

Данная команда устанавливает ограничение скорости входящего трафика. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (no) снимает ограничение.

Синтаксис

```
rate-limit <BANDWIDTH> [BURST]
no rate-limit
```

Параметры

<BANDWIDTH> – средняя скорость трафика в кбит/с, принимает значение [1..10000000] для TengigabitEthernet интерфейсов и [1..1000000] для прочих интерфейсов и туннелей;

<BURST> – размер сдерживающего порога в кБ, принимает значение [4..16000]. По умолчанию 128 кБ.

Значение burst рассчитывается по следующей формуле: burst = 0.75 * rate-limit / 8

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-L2TP

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

Пример

```
rtt(config-if-gi)# rate-limit 1670000
```

service-policy

Данная команда используется для привязки указанной QoS-политики к конфигурируемому интерфейсу для классификации входящего (input) или приоритизации исходящего (output) трафика.

Использование отрицательной формы команды (no) удаляет привязку политики к данному интерфейсу.

Синтаксис

```
service-policy { input | output } <NAME>
service-policy dynamic { upstream | downstream | all }
no service-policy { input | output }
```

Параметры

<NAME> – имя QoS-политики, задаётся строкой до 31 символа;

dynamic – привязка динамической QoS-политики к конфигурируемому интерфейсу для приоритизации восходящего (upstream), нисходящего (downstream) или всего (all) трафика. Политика наполняется данными, полученными от RADIUS-сервера.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-SUBTUNNEL (только с ключом "input")

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

Пример

```
rtt(config-if-gi)# service-policy input input_policy
```

service-policy

Данной командой привязывается политика QoS к классу для создания иерархического QoS.

Использование отрицательной формы команды (no) удаляет привязку политики к классу.

Синтаксис

```
[no] service-policy <NAME>
```

Параметры

<NAME> – имя политики, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# service-policy input_policy
```

set class-default cos

Данной командой задается значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set class-default cos <COS>
no set class-default cos
```

Параметры

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

Пример

```
rtt(config-policy-map)# set class-default cos 5
```

set class-default dscp

Данной командой задается значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set class-default dscp <DSCP>
no set class-default dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

Пример

```
rtt(config-class-map)# set class-default dscp 16
```

set class-default ip-precedence

Данной командой задается значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих классу по умолчанию (class-default).

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set class-default ip-precedence <IPP>
no set class-default ip-precedence
```

Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

Пример

```
rtt(config-class-map)# set class-default ip-precedence 5
```

set cos

Данной командой задается значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set cos <COS>
no set cos
```

Параметры

<COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-map)# set cos 5
rtt(config-class-policy-map)# set cos 5
```

set dscp

Данной командой задается значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set dscp <DSCP>
no set dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения [0..63].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-map)# set dscp 16
rtt(config-class-policy-map)# set dscp 5
```

set ip-precedence

Данной командой задается значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set ip-precedence <IPP>
no set ip-precedence
```

Параметры

<IPP> – значение кода IP Precedence, принимает значения [0..7].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-map)# set ip-precedence 5
rtt(config-class-policy-map)# set ip-precedence 5
```

set queue

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой задается номер выходной очереди QoS, в которую будут направлены пакеты, соответствующие конфигурируемому классу.

Использование отрицательной формы команды (no) отменяет назначение.

Синтаксис

```
set queue <QUEUE>
no set queue
```

Параметры

<QUEUE> – номер выходной аппаратной очереди QoS, принимает значения [1..8].

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-MAP

Пример

```
rtt(config-class-map)# set queue 5
```

shape auto-distribution

Данной командой включается автоматическое распределение полосы пропускания между классами, в которых нет настройки полосы пропускания, включая класс по умолчанию.

Использование отрицательной формы команды (no) отключает автоматическое распределение полосы.

Синтаксис

```
[no] shape auto-distribution
```

Параметры

Команда не имеет параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

Пример

```
rtt(config-policy-map)# shape auto-distribution
```

shape average

Данной командой устанавливается ограниченная и гарантированная полоса исходящего трафика для определенного класса или политики в целом.

При использовании команды совместно с командой *shape peak* для взвешенного класса полоса трафика ограничивается командой *shape peak*.

Использование отрицательной формы команды (no) снимает ограничение.

Синтаксис

```
shape average { <BANDWIDTH> | percent <BANDWIDTH_PERCENT> } [<BURST>]
no shape average
```

Параметры

<BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [1..10000000];

<BURST> – размер сдерживающего порога в байтах, принимает значение: [128..16000000];

<BANDWIDTH_PERCENT> – гарантированная полоса трафика в %, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):

- значения shape average корневой политики;
- значения traffic-shape на сетевом интерфейсе, bridge, туннеле;
- значения speed сетевого интерфейса.

Принимает значение [1..100].

Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128000.

Необходимый уровень привилегий

10

Командный режим

CONFIG-POLICY-MAP

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-policy-map)# shape average 100000 2000
```

shape peak

Устанавливается разделяемая полоса исходящего трафика для определенного класса. Данную полосу класс может занять, если другие классы не заняли свою гарантированную полосу. Конфигурация актуальна только для классов работающих во взвешенном режиме (WRR).

Использование отрицательной формы команды (no) снимает ограничение.

Синтаксис

```
shape peak { <BANDWIDTH> | percent <PERCENT> } [<BURST>]
no shape peak
```

Параметры

<BANDWIDTH> – разделяемая полоса трафика в кбит/с, принимает значение [1..10000000];

<BURST> – размер сдерживающего порога в байт, принимает значение: [128..16000000];

<BANDWIDTH_PERCENT> – общая для priority class полоса трафика в %, конкуренция происходит на основании приоритета класса, рассчитывается от (в порядке от более приоритетного к менее приоритетному значению):

- значения shape average корневой политики;
- значения traffic-shape на сетевом интерфейсе, bridge, туннеле;
- значения speed сетевого интерфейса.

Принимает значение [1..100].

Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128000.

Необходимый уровень привилегий

10

Командный режим

CONFIG-CLASS-POLICY-MAP

Пример

```
rtt(config-class-policy-map)# shape average 100000 2000
```

show qos interface shapers

Данная команда показывает параметры QoS сетевых интерфейсов.

Синтаксис

```
show qos interface shapers <IF>
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos interface shapers gigabitethernet 1/0/2
gigabitethernet 1/0/2
Committed rate:  100000 Kbps
Committed burst: 1600 KBytes
```

show qos map cos-queue

Данная команда показывает информацию о соответствии кодов 802.1p в пакетах и выходных очередей, используемых в QoS.

Синтаксис

```
show qos map cos-queue
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos map dscp-queue
d1 : d2    0  1  2  3  4  5  6  7
-------    ------------------------
0          01 02 03 04 05 06 07 08
```

show qos map dscp-mutation

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и кодов DSCP после изменений.

Синтаксис

```
show qos map dscp-mutation
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos map dscp-mutation
d1 : d2   0  1  2  3  4  5  6  7  8  9
-------   ------------------------------
0         00 01 02 03 04 05 06 07 08 09
1         10 11 12 13 14 15 16 17 18 19
2         20 21 22 23 24 25 26 27 28 29
3         30 31 32 33 34 35 36 37 38 39
4         40 41 42 43 44 45 46 47 48 49
5         50 51 52 53 54 55 56 57 58 59
6         60 61 62 63
```

show qos map dscp-queue

Данная команда показывает информацию о соответствии кодов DSCP в пакетах и выходных очередей, используемых в QoS.

Синтаксис

```
show qos map dscp-queue
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos map dscp-queue
d1 : d2    0  1  2  3  4  5  6  7  8  9
-------    ------------------------------
0          01 01 01 01 01 01 01 01 02 02
1          02 02 02 02 02 02 03 03 03 03
2          03 03 03 03 04 04 04 04 04 04
3          04 04 05 05 05 05 05 05 05 05
4          06 06 06 06 06 06 06 06 07 07
5          07 07 07 07 07 07 08 08 08 08
6          08 08 08 08
```

show qos policy binding

Данная команда используется для просмотра привязанной QoS-политики к заданному интерфейсу для классификации входящего (input) или приоритизации исходящего (output) трафика.

Синтаксис

```
show qos policy binding [ <IF> | <TUN> ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos policy binding gi1/0/1
gigabitethernet 1/0/1
Output: parent
    Class: class1
        Policy: child1
            Class: class3
    Class: class2
        Policy: child2
```

show qos policy configuration

Данная команда выводит конфигурацию заданной политики QoS.

Синтаксис

```
show qos policy configuration <NAME>
```

Параметры

<NAME> – имя политики, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos policy configuration parent
Policy-map: parent
    Shape average:
        Bandwidth:          --
        Burst:              128
    Class: class1
        Set:
            COS:            --
            DSCP:           --
            IP-Precedence:  --
            Queue:          --
        Match access group: --
        Shape average:
            Bandwidth:      --
            Burst:          128
        Shape peak:
            Bandwidth:      --
            Burst:          128
        Mode:               FIFO
        Priority class:     8
        Queue limit:        127
        Service policy:     child1
    Class: class2
        Set:
            COS:            --
            DSCP:           --
            IP-Precedence:  --
            Queue:          --
        Match access group: --
        Shape average:
            Bandwidth:      --
            Burst:          128
        Shape peak:
            Bandwidth:      --
            Burst:          128
        Mode:               FIFO
        Priority class:     8
        Queue limit:        127
        Service policy:     child2
```

show qos policy statistics

Данная команда выводит статистику по переданным и отброшенным пакетам согласно политике, назначенной на output. Команда актуальна только для Policy-based QoS режима интерфейса.

Синтаксис

```
show qos policy statistics [ <IF> | <TUN> ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt#  sh qos policy statistics
gigabitethernet 1/0/2
   Policy pom-1
   Input policy root
   Input class root
   Shape: bytes 750947679, packets 496667, drops 1002200
       Class clm-1
           Shape: bytes 750946896, packets 496658, drops 1002200
       Class class-default
           Shape: bytes 783, packets 9, drops 0
```

show qos statistics

Данная команда выводит статистику по переданным и отброшенным пакетам. Команда актуальна только для basic QoS режима интерфейса.

Синтаксис

```
show qos statistics [ <IF> | <TUN> ]
```

Параметры

<IF> – интерфейс или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos statistics vti 2
vti 2
Queue   Bytes                   Packets                 Drops
-----   --------------------    --------------------    --------------------
1       0                       0                       0
2       0                       0                       0
3       0                       0                       0
4       0                       0                       0
5       0                       0                       0
6       964073836       1413598          0
7       121389180       177990            1235497
8       0                       0                       0
```

show qos tunnel shapers

Данная команда показывает параметры QoS-туннелей.

Синтаксис

```
show qos tunnel shapers <TUN>
```

Параметры

<TUN> – имя туннеля, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show qos tunnel shapers vti 2
vti 2
qid   Target        Target
      Committed     Committed
      Rate [Kbps]   Burst [KBytes]
---   -----------   --------------
1     10000         128
2     6000          128
```

traffic-shape

Данная команда устанавливает ограничение скорости исходящего трафика для определенной очереди или интерфейса в целом. Команда актуальна только для Basic QoS режима интерфейса.

Использование отрицательной формы команды (no) снимает ограничение.

Синтаксис

```
traffic-shape { <BANDWIDTH> [BURST] | queue <QUEUE> <BANDWIDTH> [BURST] }
no traffic-shape [ queue <QUEUE> ]
```

Параметры

<QUEUE> – идентификатор очереди, принимает значение [1..8];

<BANDWIDTH> – средняя скорость трафика в кбит/с, принимает значения:

- для интерфейсов gigabitethernet, loopback, e1: [1..10000000];
- для интерфейсов tengigabitethernet: [3000..100000000];
- для интерфейсов loopback:

<BURST> – размер сдерживающего порога в кБ, принимает значение [4..16000]. Возможно указывать значение только кратное 4.

Значение по умолчанию

<BANDWIDTH> – отсутствует;

<BURST> – 128.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-BRIDGE

CONFIG-LOOPBACK

CONFIG-IF-E1

CONFIG-IP4IP4

CONFIG-GRE

CONFIG-VTI

CONFIG-L2TPV3

CONFIG-PPPOE

CONFIG-PPTP

CONFIG-L2TP

CONFIG-IF-MULTILINK

CONFIG-OPENVPN

Пример

```
rtt(config-if)# traffic-shape queue 3 100000 2000
```

27. МОНИТОРИНГ И УПРАВЛЕНИЕ

27.1. Управление Netflow

ip netflow export

Данная команда используется для включения экспорта статистики Netflow на сетевом интерфейсе.

Использование отрицательной формы команды (no) отключает экспорт статистики Netflow на сетевом интерфейсе.

Синтаксис

```
ip netflow export [ <DIRECTION> ]
no ip netflow export
```

Параметры

<DIRECTION> – направление трафика, по которому собирается статистика, принимает значения:

- ingress – статистика собирается только для входящего трафика;
- egress – статистика собирается только для исходящего трафика;
- both – статистика собирается для входящего и исходящего трафика.

Значение по умолчанию

both

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LT

CONFIG-PPPOE

Пример

```
rtt(config-if-gi)# ip netflow export
```

netflow active-timeout

Данной командой задаётся интервал, по истечении которого информация об активных сессиях экспортируются на коллектор.
В случае если значение параметра active-timeout превышает значение параметра inactive-timeout, то вместо экспортирования будет происходить накопление данных с заданной периодичностью active-timeout.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow active-timeout <TIMEOUT>
no netflow active-timeout
```

Параметры

<TIMEOUT> – задержка перед отправкой информации об активных сессиях, задается в секундах, принимает значение [5..36000].

Значение по умолчанию

1800 секунд.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow active-timeout 30
```

netflow collector

Данная команда используется для создания коллектора Netflow и перехода в командный режим CONFIG-NETFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор Netflow.

Синтаксис

```
[no] netflow collector <ADDR> [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow collector 10.100.100.1
rtt(config-netflow-host)#
```

netflow domain-id

Данной командой устанавливается значение Observation Domain ID в пакетах Netflow.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow domain-id <ID>
no netflow domain-id
```

Параметры

<ID> – значение Observation Domain ID, принимает значение [0.. 4294967295].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# netflow domain-id 35
```

netflow enable

Данной командой активируется Netflow на маршрутизаторе.

Использование отрицательной формы команды (no) деактивирует Netflow на маршрутизаторе.

Синтаксис

```
[no] netflow enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Процесс выключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow enable
```

netflow export-events web-proxy

Данная команда позволяет включить передачу netflow-статистики о пакетах, переданных через web-proxy.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] netflow export-events web-proxy
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow export-events web-proxy
```

netflow export-options

Данная команда позволяет включить дополнительные поля в экспортируемую на коллектор информацию.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] netflow export-options { http-host | http-url | ttl}
```

Параметры

http-host – для HTTP/HTTPS-сессий будет добавляться информация о домене хоста, на который был отправлен HTTP-запрос. Поле в Netflow-пакете имеет идентификатор 24884, размерность 128 байт.

http-url – для HTTP-сессий будет добавляться информация о URL, на который был отправлен HTTP-запрос. Поле в Netflow-пакете имеет идентификатор 24887, размерность 384 байта.

ttl – Добавляет значение поля TTL/Hop Limit из IP-заголовка исходного пакета в экспортируемую NetFlow-запись. Поле в Netflow-пакете имеет идентификатор 192, размерность 1 байт.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow export-options http-host
```

netflow inactive-timeout

Данной командой задаётся интервал, по истечении которого информация об устаревших сессиях экспортируется на коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow inactive-timeout <TIMEOUT>
no netflow inactive-timeout
```

Параметры

<TIMEOUT> – задержка перед отправкой информации об устаревших сессиях, задается в секундах, принимает значение [0..240].

Значение по умолчанию

15

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow inactive-timeout 30
```

netflow max-flows

Данной командой задаётся максимальное количество наблюдаемых сессий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow max-flows <COUNT>
no netflow max-flows
```

Параметры

<COUNT> – количество наблюдаемых сессий, принимает значение [10000..2000000].

Значение по умолчанию

512000

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# netflow max-flows 300000
```

netflow refresh-rate

Данной командой задаётся частота отправки шаблонов на Netflow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow refresh-rate <RATE>
no netflow refresh-rate
```

Параметры

<RATE> – частота отправки статистики, задается в пакетах на поток, принимает значение [1..10000].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# netflow refresh-rate 100
```

netflow version

Данной командой задаётся версия Netflow-протокола.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow version <VERSION>
no netflow version
```

Параметры

<VERSION> – версия Netflow-протокола: 5, 9 и 10.

Значение по умолчанию

9

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow version 10
```

netflow self-ifindex

Данной командой устанавливается ifindex для трафика, принадлежащего маршрутизатору, или для трафика, который был отброшен в результате обработки маршрутизатором. В результате выполнения команды в статистике, отправленной на коллектор, ifindex для данного вида трафика будет соответствовать настроенному.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
netflow self-ifindex <INDEX>
no netflow version
```

Параметры

<INDEX> – 4294967295 или 0.

Значение по умолчанию

4294967295

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# netflow self-ifindex 0
```

port

Данной командой определяется порт Netflow-сервиса на сервере сбора статистики.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

Значение по умолчанию

2055

Необходимый уровень привилегий

15

Командный режим

CONFIG-NETFLOW-HOST

Пример

```
rtt(config-netflow-host)# port 5555
```

show netflow configuration

Командой выполняется просмотр параметров конфигурации Netflow-агента.

Синтаксис

```
show netflow configuration
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show netflow configuration
Netflow configuration:
Global state:      Enabled
Version:           9
Domian ID:         0
Maxflows:          512000
Refresh rate:      10
Self ifindex:      4294967295
Active timeout:    1800
Inactive timeout:  15
Export-options:

Collectors:
    192.168.39.1:2055
```

show netflow statistics

Команда для просмотра текущей информации о работе Netflow.

Синтаксис

```
show netflow statistics
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show netflow statistics
NAT events: Enabled; count start 0; count  stop 0
Flows:      Active 0; peak (0 reached 55 seconds ago); memory 3931 Kbytes
Hash:       Size 503185 Kbytes; memory 3931 Kbytes
InHash:     Packets 0; size 0 Kbytes
InPDU:      Packets 0; size 0 bytes

Processed rate   Bits/s       Packets/s
--------------   ----------   ----------
Current          0            0
1 Min Avg        0            0
5 Min Avg        0            0
```

show netflow statistics cpu

Команда для просмотра статистики по распределению информации о нагрузке Netflow на ЦПУ.

Синтаксис

```
show netflow statistics
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show netflow statistics cpu
Export:
  Rate:         0 bytes/s
  Total:        Packets 2; size 0 Mbytes; flows 0
  Errors:       Packets 0
  Traffic lost: Packets 0; size 0 Kbytes; flows 0

CPU      Rate, Packets/s   Traffic, Packets   Traffic, MBytes   Drop, Packets   Drop, KBytes
-------   ---------------   ----------------   ---------------   -------------   -----------
0         0                 0                  0                 0               0
1         0                 0                  0                 0               0
2         0                 0                  0                 0               0
3         0                 0                  0                 0               0
-------   ---------------   ----------------   ---------------   -------------   -----------
-
Summary   0                 0                  0                 0               0
```

source-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых Netflow пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

Синтаксис

```
source-address <ADDR>
no source-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-NETFLOW-HOST

Пример

```
rtt(config-netflow-host)# source-address 10.100.100.2
```

27.2. Управление sFlow

ip sflow export

Данная команда используется для включения экспорта статистики sFlow на сетевом интерфейсе. Функция sFlow на сетевом интерфейсе может быть включена, если на интерфейсе выключена функция

Firewall (раздел ip firewall disable), в ином случае экспорт статистики sFlow настраивается в правиле Firewall (раздел action).

Использование отрицательной формы команды (no) отключает экспорт статистики sFlow на сетевом интерфейсе.

Синтаксис

```
[no] ip sflow export
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-SERIAL

CONFIG-IF-PORT-CHANNEL

CONFIG-CELLULAR-MODEM

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-VTI

CONFIG-GRE

CONFIG-IP4IP4

CONFIG-BRIDGE

CONFIG-LT

CONFIG-PPPOE

Пример

```
rtt(config-if-gi)# ip sflow export
```

port

Данной командой определяется порт sFlow-сервиса на сервере сбора статистики.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

Значение по умолчанию

6343

Необходимый уровень привилегий

15

Командный режим

CONFIG-SFLOW-HOST

Пример

```
rtt(config-sflow-host)# port 5556
```

sflow agent-ip

Данная команда используется для назначения адреса sFlow agent. Если команда не указана, то в качестве адреса sFlow agent будет использован случайный адрес из присутствующих в конфигурации.

Использование отрицательной формы команды (no) удаляет настройку адреса sFlow agent.

Синтаксис

```
sflow agent-ip <ADDR>
[no] sflow agent-ip
```

Параметры

<ADDR> – IPv4/IPv6-адрес. При использовании IPv4-адресации задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Для IPv6 задается в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# sflow agent-ip 192.168.44.18
```

sflow collector

Данная команда используется для создания коллектора sFlow и перехода в командный режим CONFIG-SFLOW-HOST.

Использование отрицательной формы команды (no) удаляет сконфигурированный коллектор sFlow.

Синтаксис

```
[no] sflow collector <ADDR> [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# sflow collector 10.100.100.1
```

```
rtt(config-sflow-host)#
```

sflow enable

Данной командой активируется sFlow на маршрутизаторе.

Использование отрицательной формы команды (no) деактивирует sFlow на маршрутизаторе.

Синтаксис

```
[no] sflow enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Процесс выключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# sflow enable
```

sflow poll-interval

Данной командой задаётся интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
sflow poll-interval <TIMEOUT>
no sflow poll-interval
```

Параметры

<TIMEOUT> – интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса, принимает значение [1..300].

Значение по умолчанию

10 секунд

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# sflow poll-interval 30
```

sflow sampling-rate

Данной командой задаётся частота отправки пакетов пользовательского трафика в неизменном виде на sFlow-коллектор.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
sflow sampling-rate <RATE>
no sflow sampling-rate
```

Параметры

<RATE> – частота отправки пакетов пользовательского трафика на коллектор, принимает значение [1.. 65535]. При значении частоты 10 на коллектор будет отправлен один пакет из десяти.

Значение по умолчанию

1000

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# sflow sampling-rate 100
```

show sflow configuration

Командой выполняется просмотр параметров конфигурации sFlow-агента.

Синтаксис

```
show sflow configuration
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show sflow configuration
sFlow configuration:
Global state:   Enabled
Sample rate:    1000
Poll interval:  10
Agent:          192.168.44.18

Collectors:
        10.100.100.1:5556
```

27.3. Управление SNMP

access

Данной командой определяется уровень доступа по протоколу SNMPv3.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
access <TYPE>
no access
```

Параметры

<TYPE> – уровень доступа:

- ro – доступ только для чтения;
- rw – доступ для чтения и записи.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# access rw
```

authentication access

Данной командой определяется режим безопасности.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
authentication access <TYPE>
no authentication access
```

Параметры

<TYPE> – режим безопасности:

- auth – используется только аутентификация;
- priv – используется аутентификация и шифрование данных.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# authentication algorithm auth
```

authentication algorithm

Данная команда определяет алгоритм аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) отключает аутентификацию.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм шифрования:

- md5 – пароль шифруется по алгоритму md5;
- sha1 – пароль шифруется по алгоритму sha1.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# authentication algorithm md5
```

authentication key

Данная команда устанавливает пароль для аутентификации SNMPv3-запросов.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
authentication key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;

encrypted – при указании команды задается зашифрованный пароль:

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# authentication key ascii-text 123456789
```

```
rtt(config-snmp-user)# authentication key ascii-text encrypted CDE65039E5591FA3F1
```

client-list

Данной командой активируется фильтрация и устанавливается профиль IP-адресов, с которых могут приниматься SNMPv3-пакеты с данным именем SNMPv3-пользователя.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

Синтаксис

```
[no] client-list <NAME>
```

Параметры

<NAME> – имя ранее сознанной object-group, задается строкой до 31 символа.

Значение по умолчанию

Ограничения отключены.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# client-list OBG005
```

community

Данной командой определяется SNMP-community для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) удаляет значение community.

Синтаксис

```
community <COMMUNITY>
no community
```

Параметры

<COMMUNITY> – сообщество для доступа по протоколу SNMP, задается строкой длинной [1..128] символа.

Значение по умолчанию

community – не задано.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-HOST

Пример

```
rtt(config-snmp-host)# community privatekey
```

ip address

Данной командой активируется фильтрация и устанавливается IP-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

Синтаксис

```
[no] ip address <ADDR>
```

Параметры

<ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Ограничения отключены.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# ip address 192.168.85.33
```

ipv6 address

Данной командой активируется фильтрация и устанавливается IPv6-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.

Использование отрицательной формы команды (no) деактивирует фильтрацию принимаемых SNMPv3-пакетов.

Синтаксис

```
[no] ipv6 address <IPV6-ADDR>
```

Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# ipv6 address AC:05:12:44::24
```

enable

Данной командой активируется SNMPv3-пользователь.

Использование отрицательной формы команды (no) деактивирует SNMPv3-пользователя.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# enable
```

oid-tree

Данной командой устанавливается OID и действие. применяемое к нему (разрешить/запретить). Более длинные OID имеют преимущество.

OID указывается в цифровой нотации.

Использование отрицательной формы команды (no) удаляет запись oid-tree.

Синтаксис

```
oid-tree <OID> <ACTION>
no oid-tree <OID>
```

Параметры

<OID> – OID, задаётся строкой до 255 символов;

<ACTION> – действие, применяемое к OID:

- excluded – запретить использование OID;
- included – разрешить использование OID.

Значение по умолчанию

Разрешено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-VIEW

Пример

```
rtt(config-snmp-view)# oid-tree 1.3.6.1.2.1.2.2 excluded
```

port

Данной командой определяется порт коллектора SNMP уведомлений на удаленном сервере.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер UDP-порта, указывается в диапазоне [1..65535].

Значение по умолчанию

162

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-HOST

Пример

```
rtt(config-snmp-host)# port 5555
```

privacy algorithm

Данная команда определяет алгоритм шифрования передаваемых данных.

Использование отрицательной формы команды (no) отключает шифрование.

Синтаксис

```
privacy algorithm <ALGORITHM>
no privacy algorithm
```

Параметры

<ALGORITHM> – алгоритм шифрования:

- aes128 – использовать алгоритм шифрования AES-128;
- aes256 – использовать алгоритм шифрования AES-256;
- des – использовать алгоритм шифрования DES.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# privacy algorithm des
```

privacy key

Данная команда устанавливает пароль для шифрования передаваемых данных.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
privacy key ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no privacy key
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 32 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 32 байт (от 16 до 64 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# privacy key ascii-text 123456789
rtt(config-snmp-user)# privacy key ascii-text encrypted CDE65039E5591FA3F1
```

rmon collection statistics

Данная команда включается сохранение RMON-статистики для физического интерфейса.

Использование отрицательной формы команды (no) отключается сохранение RMON-статистики для физического интерфейса.

Синтаксис

```
rmon collection statistics <INDEX> owner <OWNER>
no rmon collection statistics
```

Параметры

<INDEX> – RMON-индекс данного интерфейса;

<OWNER> – текстовое поле длиной [1..127] символов, описывающее владельца, создавшего данный процесс.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if)# rmon collection statistics 17 owner admin
```

snmp-server

Данной командой включается SNMP-сервер как в глобальной таблице маршрутизации, так и во всех созданных VRF.

Использование отрицательной формы команды (no) выключает SNMP-сервер.

Синтаксис

```
[no] snmp-server
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server
```

snmp-server community

Данной командой определяется сообщество для доступа по протоколу SNMP.

Использование отрицательной формы команды (no) удаляет настройки сообщества.

Синтаксис

```
[no] snmp-server community <COMMUNITY> [ <TYPE> ] [ { <ADDR> | <IPV6-ADDR> } ]
[client-list <OBJ-GROUP-NETWORK-NAME> ] [ <VERSION> ] [ view <VIEW-NAME> ] [ vrf
<VRF> ]
```

Параметры

<COMMUNITY> – сообщество для доступа по протоколу SNMP, задается строкой длиной [1..128] символа;

<TYPE> – уровень доступа:

- ro – доступ только для чтения;
- rw – доступ для чтения и записи.

<ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, от которых обрабатываются snmp-запросы, задаётся строкой до 31 символа;

<VERSION> – версия snmp, поддерживаемая данным community, принимает значения v1 или v2c;

<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID;

<VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server community public rw
```

snmp-server contact

Данной командой устанавливается значение переменной SNMP, содержащей контактную информацию (по умолчанию не определена). Для удобства в параметрах можно указать ответственного за данное оборудование, например, его фамилию.

Использование отрицательной формы команды (no) удаляет значение переменной SNMP, содержащей контактную информацию.

Синтаксис

```
[no] snmp-server contact <CONTACT>
```

Параметры

<CONTACT> – контактная информация, задаётся строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server contact ivanov_ivan
```

snmp-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SNMP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
snmp-server dscp <DSCP>
no snmp-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

61

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server dscp 40
```

snmp-server enable traps

Данная команда разрешает отправку всех типов SNMP-уведомлений.

Использование отрицательной формы команды (no) отменяет разрешение отправки всех типов SNMP-уведомлений.

Синтаксис

```
[no] snmp-server enable traps
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps
```

snmp-server enable traps config

Данная команда разрешает отправку SNMP-уведомлений об операциях с конфигурацией.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps config [ <ACT> ]
```

Параметры

<ACT> – трапы фактов изменения конфигурации:

- commit – применение изменения конфигурации;
- confirm – подтверждение изменения конфигурации.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps config commit
```

snmp-server enable traps entity

Данная команда разрешает отправку SNMP-уведомлений об операциях с running-config.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps entity [ <ENT> ]
```

Параметры

<ENT> – типы фильтров параметров окружения:

- config-change – информация о операциях с running-config.

Без указания ключа <ENT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps entity
```

snmp-server enable traps entity-sensor

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps entity-sensor [ <ENT> ]
```

Параметры

<ENT> – типы фильтров параметров окружения:

- threshold – информация о срабатывании пересечения пороговых значений.

Без указания ключа <ENT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps entity-sensor
```

snmp-server enable traps environment

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps environment [ <ENV> ]
```

Параметры

<ENV> – типы фильтров параметров окружения:

- pwrin – отказ БП;
- pwrin-insert – БП установлен;
- fan – отказ вентилятора;
- fan-speed-changed – изменение скорости вентиляторов;
- fan-speed-high – скорость вращения вентиляторов превысила максимальный порог;
- memory-flash-low – свободный объем NAND меньше заданного порога;
- memory-flash-critical-low – свободный объем NAND меньше заданного критического порога;
- memory-ram-low – свободный объем RAM меньше заданного порога;
- memory-ram-critical-low– свободный объем RAM меньше заданного критического порога;
- cpu-load – высокая нагрузка ЦПУ;
- cpu-overheat-temp – температура CPU превысила заданный максимальный порог;
- cpu-critical-temp – температура CPU превысила заданный критический порог;
- cpu-supercooling-temp – температура CPU упала ниже заданного минимального порога;
- switch-overheat-temp – температура коммутатора превысила заданный максимальный порог;
- switch-supercooling-temp – температура коммутатора упала ниже заданного минимального порога;
- board-overheat-temp – перегрев платы;
- board-supercooling-temp – переохлаждение платы;
- sfp-overheat-temp – перегрев sfp-модуля;
- sfp-supercooling-temp – переохлаждение sfp-модуля.

Без указания ключа <ENV> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps enviroment pwrin
```

snmp-server enable traps envmon

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps envmon [ <ENV> ]
```

Параметры

<ENV> – типы фильтров параметров окружения:

- fan – информация о работе блоков вентиляторов;
- shutdown – информация о отключении маршрутизатора;
- supply – информация о работе блоков питания;
- temperature информация о работе температурных датчиков.

Без указания ключа <ENV> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps envmon fun
```

snmp-server enable traps files-operations

Данная команда разрешает отправку SNMP-уведомлений об операциях с файлами.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps files-operations [ <ACT> ]
```

Параметры

<ACT> – типы фильтров параметров операций с файлами:

- successful – успешно;
- failed – неудачно;
- canceled – отменено.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps files-operations canceled
```

snmp-server enable traps flash

Данная команда разрешает отправку SNMP-уведомлений об операциях с внешними flash-накопителями.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps flash [ <ACT> ]
```

Параметры

<ACT> – типы фильтров параметров операций с файлами:

- insertion – подключение flash-накопителя;
- removal – удаление flash-накопителя.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps flash removal
```

snmp-server enable traps interfaces

Данная команда разрешает отправку SNMP-уведомлений о изменении состояния интерфейсов.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps interfaces [ <ACT> ]
```

Параметры

<ACT> – типы фильтров параметров окружения:

- rx-utilization-high – поток входящих данных превышает порог;
- tx-utilization-high – поток исходящих данный превышает порог;
- number-high – превышение количества IP-интерфейсов;

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps interfaces rx-utilization-high
```

snmp-server enable traps ports

Данная команда разрешает отправку SNMP-уведомлений о появлении ошибок на интерфейсах чипа коммутации.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps ports [ <TYPE> ]
```

Параметры

<TYPE> – типы фильтров состояние порта:

- port-counters-errors – ошибки на интерфейсах чипа коммутации.

Без указания ключа <TYPE> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps ports
```

snmp-server enable traps screens

Данная команда разрешает отправку SNMP-уведомлений о срабатывании защиты от определенного вида DoS-атак.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps screens [ <SCREEN> ]
```

Параметры

<SCREEN> – типы фильтров защиты от DoS-атак:

- dest-limit – ограничение количества одновременных сессий на основании адреса назначения;
- source-limit – ограничение количества одновременных сессий на основании адреса источника;
- icmp-threshold – защита от ICMP flood атак;
- udp-threshold – защита от UDP flood атак;
- syn-flood – защита от SYN flood атак;
- land – защита от land атак;
- winnuke – защита от winnuke атак;
- icmp-frag – блокировка фрагментированных ICMP-пакетов;
- syn-flag – блокировка фрагментированных TCP-пакетов, с флагом SYN;
- unknown-proto – блокировка пакетов, с ID протокола в заголовке IP равном 137 и более;
- ip-frag – блокировка фрагментированных пакетов;
- port-scan – защита от port scan атак;
- ip-sweep – защиту от IP-sweep атак;
- syn-fin – блокировка TCP-пакетов, с установленными флагами SYN и FIN;
- fin-no-ack – блокировка TCP-пакетов с установленным флагом FIN и не установленным флагом ACK;
- no-flag – блокировка TCP-пакетов, с нулевым полем flags;
- spoofing – защита от IP spoofing атак;
- reserved – блокировка всех ICMP-пакетов 2 и 7 типов (reserved);
- quench – блокировка всех ICMP-пакетов 4 типа (source quench);
- echo-request – блокировка всех ICMP пакетов 8 типа (echo-request);
- time-exceeded – блокировка всех ICMP-пакетов 11 типа (time exceeded);
- unreachable – блокировка всех ICMP-пакетов 3 типа (destination-unreachable);
- icmp-large – блокировка ICMP-пакетов большого объема;
- tcp-all-flags – блокировка tcp-пакетов с флагами;
- udp-frag – блокировка udp-пакетов с флагами.

Без указания ключа <LINK> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps screens reserved
```

snmp-server enable traps snmp

Данная команда разрешает отправку SNMP-уведомлений об изменении параметров окружения.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps snmp [ <ACT> ]
```

Параметры

<ACT> – типы фильтров параметров окружения:

- authentication – уведомления о snmp-запросах на маршрутизатор с неверными community или snmpv3-паролем;
- coldstart – уведомления о перезапуске snmp-сервера на маршрутизаторе;
- linkdown – информация о изменении состояния link в down;
- linkup – информация о изменении состояния link в up.

Без указания ключа <ACT> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps snmp linkup
```

snmp-server enable traps syslog

Данная команда разрешает отправку SNMP-уведомлений с syslog-сообщениями.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps syslog
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps syslog
```

snmp-server enable traps wifi

Данная команда разрешает отправку SNMP-уведомлений с сообщениями о работе Wi-Fi контроллера.

Использование отрицательной формы команды (no) отменяет разрешение отправки указанных типов уведомлений.

Синтаксис

```
[no] snmp-server enable traps wifi [ <NAME> ]
```

Параметры

<NAME> – типы трапа о туннелях softgre:

- wifi-tunnels-number-in-bridge-high – включение трапов о превышении количества sub-gre-тунелей включенных в bridge;
- wifi-tunnels-operation – включение трапов о результате snmp-операций с туннелями softgre.

Без указания ключа <NAME> – активируется отправка всех трапов данной группы.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server enable traps syslog
```

snmp-server host

Данной командой включается передача SNMP-уведомлений на указанный IP-адрес и осуществляется переход в режим настройки SNMP-уведомлений.

Использование отрицательной формы команды (no) отключает передачу уведомлений на указанный коллектор SNMP-уведомлений.

Синтаксис

```
[no] snmp-server host { <ADDR> | <IPV6-ADDR> } [vrf <VRF>]
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<VRF> – имя экземпляра VRF, в котором находится коллектор SNMP-уведомлений, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp host 192.168.2.2
```

snmp-server location

Данной командой устанавливается значение переменной SNMP, содержащей информацию о расположении оборудования (по умолчанию не определено). Для удобства в параметрах можно указать город, улицу, район, номер комнаты и т.п.

Использование отрицательной формы команды (no) удаляет значение переменной, содержащей информацию о расположении оборудования.

Синтаксис

```
[no] snmp-server location <LOCATION>
```

Параметры

<LOCATION> – информация о расположении оборудования, задаётся строкой до 255 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server location duglasa_adamsa_42
```

snmp-server trap link

Данной командой устанавливается режим отправки SNMP-trap.

Использование отрицательной формы команды (no) устанавливает режим по умолчанию.

Синтаксис

```
snmp-server trap link <MODE>
no snmp-server host
```

Параметры

<MODE> – режим отправки SNMP-trap. Принимает значения:

- ietf;
- cisco.

Значение по умолчанию

ietf

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server trap link cisco
```

snmp-server user

Данной командой создается SNMPv3-пользователь.

Использование отрицательной формы команды (no) удаляет SNMPv3-пользователя.

Синтаксис

```
[no] snmp-server user <NAME>
```

Параметры

<NAME> – имя пользователя, задаётся строкой от 1 до 128 символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server user admin
rtt(config-snmp-user)#
```

snmp-server view

Данной командой создается профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для community (SNMPv2) и user (SNMPv3).

Использование отрицательной формы команды (no) удаляет профиль snmp view.

Синтаксис

```
[no] snmp-server view <VIEW-NAME>
```

Параметры

<VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# snmp-server view user_access
rtt(config-snmp-view)#
```

source-address

Данной командой определяется IP-адрес для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no source-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.

Значение по умолчанию

IPv4/IPv6 – адрес интерфейса, ближайшего к удаленному SNMP-серверу.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-HOST

Пример

```
rtt(config-snmp-host)# source-address 192.168.22.17
```

source-interface

Данной командой определяется интерфейс или туннель маршрутизатора, IPv4/IPv6-адрес которого будет использоваться для отправки уведомлений на удаленный сервер.

Использование отрицательной формы команды (no) удаляет указанный интерфейс или туннель.

Синтаксис

```
source-interface { <IF> | <TUN> }
```

```
no source-interface
```

Параметры

<IF> – имя интерфейса устройства, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-HOST

Пример

```
rtt(config-snmp-host)# source-interface gigabitethernet 1/0/1
```

view

Данной командой устанавливается профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для SNMPv3 user.

Использование отрицательной формы команды (no) удаляет профиль snmp view.

Синтаксис

```
[no] view <VIEW-NAME>
```

Параметры

<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SNMP-USER

Пример

```
rtt(config-snmp-user)# view user_view
```

27.4. Управление SYSLOG

logging aaa configuration

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации разделов AAA.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации разделов AAA.

Синтаксис

```
[no] logging aaa configuration
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging aaa configuration
```

logging acl configuration

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации ACL.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации ACL.

Синтаксис

```
[no] logging acl configuration
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging acl configuration
```

logging firewall configuration

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации межсетевого экрана.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации межсетевого экрана.

Синтаксис

```
[no] logging firewall configuration
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging firewall configuration
```

logging login on-failure

Данной командой включается запись в локальный syslog-сервер сообщений о неуспешных попытках подключения к CLI.

Использование отрицательной формы команды (no) выключает логирование неудачных подключений.

Синтаксис

```
[no] logging login on-failure
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging login on-failure
```

logging nat

Данной командой включается запись в локальный syslog-сервер сообщений о работе сервиса NAT.

Использование отрицательной формы команды (no) выключает логирование сервиса NAT.

Синтаксис

```
[no] logging nat [<NAT-TYPE>]
```

Параметры

<NAT-TYPE> – тип сервиса NAT подлежащий логированию:

- destination;
- proxy;
- source.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging nat source
```

logging service start-stop

Данной командой включается запись в локальный syslog-сервер сообщений о запуске и остановке используемых сервисов.

Использование отрицательной формы команды (no) выключает логирование запуска и остановки используемых сервисов.

Синтаксис

```
[no] logging service start-stop
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging service start-stop
```

logging syslog configuration

Данной командой включается запись в локальный syslog-сервер сообщений о изменении конфигурации syslog-сервера.

Использование отрицательной формы команды (no) выключает логирование сообщений о изменении конфигурации syslog-сервера.

Синтаксис

```
[no] logging syslog configuration
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging syslog configuration
```

logging userinfo

Данной командой включается запись в локальный syslog-сервер сообщений о изменении user-profile.

Использование отрицательной формы команды (no) выключает логирование сообщений об изменении user-profile.

Синтаксис

```
[no] logging userinfo
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# logging userinfo
```

match process-name

Данной командой активируется или деактивируется сохранение в локальный syslog-файл или отправка на удаленный syslog-сервер событий работы отдельных процессов маршрутизатора.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
match [not] process-name <PROCESS-NAME>
no match process-name { <PROCESS-NAME> | all }
```

Параметры

<PROCESS-NAME> – имя процесса события, которое необходимо или не нужно сохранять в локальный syslog-файл или отправлять на удаленный syslog-сервер. Принимает значения:

- aaa-mgr – вывод информационных сообщений работы сервисов AAA на устройстве;
- address-resolving – вывод информационных сообщений сервиса предобразования DNS-имен в IP-адреса;
- alarm-mgr – вывод информационных сообщений менеджера аварийных событий устройства;
- avahi – вывод информационных сообщений сервиса mDNS;
- cfgsync-mgr – вывод информационных сообщений менеджера конфигурации устройства (применения/отката/сохранения/загрузки);
- cli – вывод информационных сообщений командной оболочки CLI;
- cluster-mgr - вывод информационных сообщений менеджера отвечающего за работу кластера;
- device-mgr – вывод информационных сообщений подключаемых устройств (USB-flash,mmc,SD,HDD);
- dhcp-client – вывод информационных сообщений DHCP-клиента;
- dhcp-proxy – вывод информационных сообщений DHCP-proxy;
- dhcp-server – вывод информационных сообщений DHCP-сервера;
- dns – вывод информационных сообщений сервиса DNS;
- e1d – вывод информационных сообщений E1/Multilink;
- env-mgr – вывод информационных сообщений мониторинга аппаратных параметров окружения системы (БП/FAN/RAM/FLASH/CPU);
- rttfs – вывод информационных сообщений, связанных с файловой системой устройства (чтение, запись);
- file-mgr – вывод информационных сообщений при работе с файлами (операция copy);
- firewall-failover – вывод информационных сообщений сервиса Firewall-failover;
- firewalld – вывод информационных сообщений сервиса Firewall;
- haveged – вывод информационных сообщений службы генератора случайных чисел устройства;

- if-mgr – вывод информационных сообщений менеджера интерфейсов (состояние, сбор статистики);
- if-mgr-ng – вывод информационных сообщений менеджера интерфейсов (конфигурирование и перенастройка);
- ipc-hub – вывод информационных сообщений межпроцессного взаимодействия между различными сервисами системы;
- ipc-mgr – логирование событий IPC-шины устройства;
- ips – вывод информационных сообщений отработки правил IPS/IDS;
- ips-storage-mgr – вывод информационных сообщений хранения IPS/IDS-сигнатур;
- ipsec – вывод информационных сообщений сервиса IPsec;
- key-safed – вывод информационных сообщений сервиса безопасного хранения криптоключей на внешних носителях и подключения/отключения криптохриптохранилищ на внешних носителях;
- l2tpclient – вывод информационных сообщений L2TP-клиента;
- lb – вывод информационных сообщений сервиса балансировки трафика на CPU;
- lic-mgr – вывод информационных сообщений сервиса лицензирования;
- lldpd – вывод информационных сообщений LLDP;
- modem-cfgmgr – вывод информационных сообщений конфигурирования (параметры, применение, конфликт конфигурации) USB-модемов на маршрутизаторе;
- modem-mgr – логирование низкоуровневых процессов подключения USB-модемов к устройству (идентификация типа и дальнейшее подключение);
- multiwan – вывод информационных сообщений MultiWAN;
- nhrp – вывод информационных сообщений NHRP;
- ntp – вывод информационных сообщений NTP;
- oi-mgr – вывод информационных сообщений менеджера оперативной информации внутрисистемных процессов устройства;
- openvpn – вывод информационных сообщений OpenVPN;
- pci-boot - вывод информационных сообщений взаимодействия двух процессоров при загрузке
- pppoe – вывод информационных сообщений PPPoE;
- pptp – вывод информационных сообщений PPTP;
- rexd – вывод информационных сообщений работы регулярных выражений (компиляция, проверка на совпадения и отработка);
- rngd – вывод информационных сообщений службы генераторы случайных чисел устройства;
- routing – вывод информационных сообщений процесса маршрутизации;
- service-mgr – вывод информационных сообщений менеджера вторичных сервисов устройства;
- session-mgr – вывод информационных сообщений менеджера регистрации и мониторинга CLI-сессий;
- sflowd – вывод информационных сообщений сервиса sFlow;
- sla-mgr – вывод информационных сообщений работы тестов IP SLA;
- snmp-server – вывод информационных сообщений SNMP;
- ssh-server – вывод информационных сообщений SSH-сервера;
- sync-mgr - вывод информационных сообщений процесса синхронизации;
- switch – вывод информационных сообщений работы аппаратного чипа коммутации (только на R800);
- sync-mgr – вывод информационных сообщений синхронизации служебных данных системы на удалённых устройствах;
- syslog-mgr – вывод информационных сообщений менеджера логирования (выборка процессов логирования, классифицирование по необходимому уровню логирования);
- syslog-ng – вывод информационных сообщений менеджера логирования (отправка сгенерированных сообщений в необходимый интерфейс вывода данных);
- systemdb – вывод информационных сообщений менеджера аварий сервисов системы;

- telnet-server – вывод информационных сообщений Telnet-сервера;
- vrrp – вывод информационных сообщений VRRP;
- wireguard – вывод информационных сообщений Wireguard;
- zabbix-agent – вывод информационных сообщений Zabbix-агента;
- zabbix-proxy – вывод информационных сообщений Zabbix-прокси сервера;
- others – логирование прочих процессов, не попадающих под какую-либо классификацию.

all – ключ для удаление всех ранее сконфигурированных процессов.

Значение по умолчанию

Сохраняются/отправляются события всех процессов.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SYSLOG-SNMP

CONFIG-SYSLOG-HOST

CONFIG-SYSLOG-FILE

Пример

```
rtt(config-syslog-file)# match not process-name dhcp-client
```

ntp logging

Данной командой включается запись в локальный syslog-сервер сообщений о работе с NTP-пирами.

Использование отрицательной формы команды (no) выключает логирование сообщений о работе с NTP-пирами.

Синтаксис

```
[no] ntp logging
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ntp logging
```

port

Данной командой указывается номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port <PORT>
no port
```

Параметры

<PORT> – номер TCP/UDP-порта, на который будут отправляться пакеты с syslog-сообщениями.

Значение по умолчанию

514

Необходимый уровень привилегий

10

Командный режим

CONFIG-SYSLOG-HOST

Пример

```
rtt(config-syslog-host)# port 5014
```

remote-address

Данной командой указывается IPv4/IPv6-адрес удаленного syslog-сервера.

Использование отрицательной формы команды (no) удаляет IPv4/IPv6-адрес удаленного syslog-сервера.

Синтаксис

```
remote-address { <ADDR> | <IPV6-ADDR> }
no remote-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SYSLOG-HOST

Пример

```
rtt(config-syslog-host)# remote-address 192.168.0.100
```

severity

Данной командой указывается уровень важности сообщений, которые будут сохраняться в локальный syslog-файл или отправляться на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
severity <SEVERITY>
no severity
```

Параметры

<SEVERITY> – уровень важности сообщения, принимает значения (в порядке убывания важности):

- none – отключает вывод syslog-сообщений в консоль;
- emerg – в системе произошла критическая ошибка, система неработоспособна;
- alert – сигналы тревоги, необходимо немедленное вмешательство персонала;
- crit – критическое состояние системы, сообщение о событии;
- error – сообщения об ошибках;
- warning – предупреждения, неаварийные сообщения;
- notice – сообщения о важных системных событиях;
- info – информационные сообщения системы;

- debug – отладочные сообщения, предоставляют пользователю информацию для корректной настройки системы.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SYSLOG-SNMP

CONFIG-SYSLOG-HOST

CONFIG-SYSLOG-FILE

CONFIG-SYSLOG-CONSOLE

CONFIG-SYSLOG-MONITOR

Пример

```
rtt(config-syslog-file)# severity info
```

show syslog

Команда для просмотра текущей информации о конфигурации syslog-журнала, списка созданных log-файлов, а также для просмотра log-файлов с возможностью фильтрации с помощью регулярных выражений.

Синтаксис

```
show syslog <FILE> [ from-date <YEAR> <MONTH> <DAY> ] [ from-time <TIME> ] [ to-date <YEAR> <MONTH> <DAY> ] [ to-time <TIME> ] [ from-end ]
```

Параметры

<FILE> – имя файла, задаётся строкой до 31 символа;

from-date – для вывода информации, начиная с указываемой даты;

from-time – для вывода информации, начиная с указываемого времени;

to-date – для вывода информации до указываемой даты;

to-time – для вывода информации до указываемого времени;

<YEAR> – год, принимает значения [2001..2037].

<MONTH> – месяц, принимает значения [January/February/March/April/May/June/July/August/September/October/November/December];

<DAY> – день месяца, принимает значения [1..31];

<TIME> – устанавливаемое системное время, задаётся в виде HH MM SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0 .. 59];
- SS – секунды, принимает значение [0 .. 59].

from-end – просмотр содержимого файла с конца, так как последние записи помещаются в конец файла.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show syslog
Name                                    Size            Date of last modification
--------------------------------------  --------  --    -------------------------
flash:critlog/log                       113.48    KB    2024-06-27 10:14:50
flash:critlog/log.1                     499.99    KB    2023-12-17 02:33:04
flash:critlog/reload_reason.txt         78.00     B     2024-06-06 10:01:27
tmpsys:syslog/auth.log                  62.11     KB    2024-07-02 08:33:20
Total files: 4
rtt# show syslog configuration
SYSLOG
File size: 512 (kiB)
Number of logs: 3
Console: info
   Files:
   ~~~~~~
ID   Name                              Severity
--   --------------------------------  ----------
0    rtt                               info
```

show syslog configuration

Команда для просмотра текущей информации о конфигурации syslog-журнала.

Синтаксис

```
show syslog configuration [ console | files [<FILE>] | firewall | hosts [<NAME>] |
monitor | snmp ]
```

Параметры

console – конфигурация syslog-событий при консольном подключении;

files – информация о файлах для записи syslog-событий;

firewall – конфигурация syslog-событий межсетевого экрана;

hosts – вывод информации о хостах, на которые передаются syslog-журналы;

monitor – вывод конфигурации монитора syslog-журнала;

snmp – вывод конфигурации snmp syslog-журнала.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show syslog configuration
Syslog:
    Write sequence numbers:             Disabled
    Write msec in timestamp:            Disabled
    Write program names:                Disabled

    Log syslog configuration changes:   Disabled
    Debug severity on reload:           Disabled

Console:
    Console severity:                   info
    Console filters defined:            Disabled

Monitor:
    Monitor severity:                   info
    Monitor filters defined:            Disabled

User Info:
    Log AAA configuration changes:      Disabled
    Log ACL configuration changes:      Disabled
    Log user info changes:              Disabled
    Log user commands:                  Disabled
    Log failed authentication attempts: Disabled

IPS:
    IPS severity:                       info

SNMP:
    SNMP trap status:                   Disabled
    SNMP severity:                      info
    SNMP filters defined:               Disabled

Network:
    Log IPv4 BFD neighbor changes:      Disabled
    Log IPv4 BGP neighbor changes:      Disabled
    Log IPv4 OSFP neighbor changes:     Disabled
```

```
    Log IPv6 BFD neighbor changes:      Disabled
    Log IPv6 BGP neighbor changes:      Disabled
    Log IPv6 OSFP neighbor changes:     Disabled

    Log NAT source:                     Disabled
    Log NAT destination:                Disabled
    Log HTTP proxy:                     Disabled

    NTP peer logging:                   Disabled

SLA:
    Log SLA delay:                      Disabled
    Log SLA error:                      Disabled
    Log SLA jitter:                     Disabled
    Log SLA losses:                     Disabled
    Log SLA status:                     Disabled

Firewall:
    Log firewall configuration changes: Disabled
    Log firewall screen:                Disabled
    Detailed firewall screen logging:   Disabled

  Files:
  ~~~~~~
Name                               Severity     Filters
---------------------------------  ----------   --------
tmpsys:syslog/DHCP                 Info         Yes
tmpsys:syslog/debug                Debug        No
tmpsys:syslog/NOT-DHCP             Info         Yes
tmpsys:syslog/test                 Debug        Yes

  Hosts:
  ~~~~~~
Name                      IP Address                      Severity   Transport   Port
Filters
-----------------------   -----------------------------   --------   ---------   -----   ----
----
syslog_host               192.168.1.2                     Info       UDP         514     No
```

source-address

Данной командой указывается IPv4/IPv6-адрес маршрутизатора, от которого будут отправляться пакеты на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
source-address { <ADDR> | <IPV6-ADDR> | object-group <NETWORK_OBJ_GROUP_NAME> }
no source-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address.

Значение по умолчанию

IPv4/IPv6-адрес интерфейса, с которого отправляется пакеты на удаленный syslog-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SYSLOG-HOST

Пример

```
rtt(config-syslog-host)# source-address 192.168.54.157
```

syslog cli-commands

Данной командой включается процесс логирования введённых команд пользователя на локальный syslog-сервер.

Использование отрицательной формы команды (no) выключает логирование команд.

Синтаксис

```
[no] syslog cli-commands
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog cli-commands
```

syslog console

Данной командой включается отображение syslog-сообщений при консольном подключении и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отображение сообщений по умолчанию.

Синтаксис

```
syslog console
no syslog console
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog console
```

syslog facility

Данной командой указывается категория сообщений, с которой они будут сохраняться в локальный syslog-файл или отправляться на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
syslog facility <FACILITY>
no syslog facility
```

Параметры

FACILITY	Numerical Code
local0	16
local1	17
local2	18
local3	19
local4	20
local5	21

FACILITY	Numerical Code
local6	22
local7	23

Значение по умолчанию

local7

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog facility local0
```

syslog file

Данной командой включается сохранение сообщений syslog в указанный файл и происходит переход в режим конфигурирования параметров данного syslog-файла.

Использование отрицательной формы команды (no) отключает сохранение сообщений syslog в указанный файл.

Синтаксис

```
syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> | hdd://<HDD>:/<NAME> |
mmc://<MMC>:/<NAME> | usb://<USB>:/<NAME> }
no syslog file { flash:syslog/<NAME> | tmpsys:syslog/<NAME> | hdd://<HDD>:/<NAME> |
mmc://<MMC>:/<NAME> | usb://<USB>:/<NAME> | all}
```

Параметры

flash – файл располагается в энергонезависимой памяти устройства;

tmpsys – файл располагается в энергозависимой памяти устройства;

hdd, mmc, usb – файл располагается на внешнем носителе;

<NAME> – имя файла, в который будет производиться запись сообщений заданного уровня, задается строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут отключено сохранение во все сконфигурированные syslog-файлы.

Размер файловой системы tmpsys составляет 100 МБ.

Настраивая число файлов в ротации и размер файла, необходимо следить за тем, чтобы суммарный размер файлов не превышал эти ограничения.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog file flash:syslog/file
```

syslog file-size

Командой устанавливается максимальный размер файла журнала. По превышении указанного размера будет автоматически производиться ротация файлов.

Использование отрицательной формы команды (no) устанавливает значение размера файла журнала в значение по умолчанию.

Синтаксис

```
syslog file-size <SIZE>
no syslog file-size
```

Параметры

<SIZE> – размер файла, принимает значение [10..10000000] Кбайт.

Значение по умолчанию

500 Кбайт.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog file-size 10000
```

syslog host

Данной командой включается передача сообщений syslog на удаленный syslog-сервер и происходит переход в режим конфигурирования параметров данного syslog-сервера.

Использование отрицательной формы команды (no) отключает передачу сообщений syslog на удаленный syslog-сервер.

Синтаксис

```
syslog host <HOSTNAME>
no syslog host { <HOSTNAME> | all }
```

Параметры

<HOSTNAME> – наименование syslog-сервера, задаётся строкой до 31 символа. Используется только для идентификации сервера при конфигурировании. Значение «all» используется в команде **no syslog host all** для удаления всех syslog-серверов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog host Rusteletech
```

syslog max-files

Данная команда устанавливает максимальное количество файлов, сохраняемых при ротации.

Использование отрицательной формы команды (no) устанавливает количество хранимых файлов журнала в значение по умолчанию.

Синтаксис

```
syslog max-files <NUM>
no syslog max-files
```

Параметры

<NUM> – максимальное количество файлов, принимает значения [1 .. 1000].

Значение по умолчанию

15

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog max-files 100
```

syslog monitor

Данной командой включается отображение syslog-сообщений при удаленных подключениях (Telnet, SSH) и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отображение сообщений по умолчанию.

Синтаксис

```
syslog monitor
no syslog monitor
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# syslog monitor
```

syslog program-name

Данной командой включается добавление имени процесса к записям в локальный syslog-сервер.

Использование отрицательной формы команды (no) убирает имя процесса из сообщений.

Синтаксис

```
syslog program-name
```

```
no syslog program-name
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# syslog program-name
```

syslog reload debugging

Данной командой включается режим debugging в процессе перезагрузки для локального syslog-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] syslog reload debugging
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog reload debugging
```

syslog sequence-numbers

Данной командой включается добавление порядкового номера к записям в локальный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] syslog sequence-numbers
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog sequence-numbers
```

syslog sip-level

Данной командой устанавливается уровень логирования sip-событий.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] syslog sip-level <LEVEL>
```

Параметры

<LEVEL> – уровень sip-сообщений, который будут попадать в syslog. Принимает значение в диапазоне [0..9].

Значение по умолчанию

0

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog sip-level 8
```

syslog snmp

Данной командой включается отображение syslog-сообщений на snmp-сервер в виде snmp-trap и происходит переход в режим конфигурирования параметров.

Использование отрицательной формы команды (no) устанавливает отправку сообщений по умолчанию.

Синтаксис

```
syslog snmp
no syslog snmp
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# syslog snmp
```

syslog timestamp msec

Данной командой включается добавление миллисекунд в таймштамп.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] syslog timestamp msec
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# syslog timestamp msec
```

transport

Данной командой указывается транспортный протокол для передачи пакетов на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
transport { tcp | udp }
no transport
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

udp

Необходимый уровень привилегий

10

Командный режим

CONFIG-SYSLOG-HOST

Пример

```
rtt(config-syslog-host)# transport tcp
```

vrf

Данной командой указывается имя экземпляра VRF, в рамках которого будут отправляться пакеты на удаленный syslog-сервер.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
vrf <VRF>
no vrf
```

Параметры

<VRF> – имя экземпляра VRF, в котором доступен удаленный syslog-сервер, задается строкой до 31 символа.

Значение по умолчанию

Отсутствует (глобальная таблица маршрутизации).

Необходимый уровень привилегий

10

Командный режим

CONFIG-SYSLOG-HOST

Пример

```
rtt(config-syslog-host)# vrf MGMT
```

27.5. Настройка доступа SSH, Telnet

ip ssh access-addresses

Данная команда ограничивает доступ до SSH-сервера. SSH-сервер становится доступным только с определённых адресов.

Использование отрицательной формы команды (no) разрешает доступ с любых адресов.

Данная команда позволяет ограничивать доступ до SSH-сервера при отключенных функциях Firewall или ACL.

В случае совместного использования данного функционала с функциями Firewall и ACL необходимо, чтобы трафик был разрешён всеми тремя функциями.

Синтаксис

```
ip ssh access-addresses [ vrf <NAME> ] <OBJ-GR-NAME>
no ip ssh access-addresses [ vrf <NAME> ]
```

Параметры

<OBJ-GR-NAME> – имя профиля IP-адресов, с которых разрешён доступ.

<VRF> – имя экземпляра VRF, в рамках которого будет работать ограничение доступа на SSH-сервер.

Необходимый уровень привилегий

15

Значение по умолчанию

Доступ разрешён с любых адресов.

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh access-addresses MGT
```

ip ssh authentication algorithm disable

Данная команда запрещает использование определенного алгоритма аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма аутентификации для SSH-сервера.

Синтаксис

```
[no] ip ssh authentication algorithm <ALGORITHM> disable
```

Параметры

<ALGORITHM> – алгоритм хеширования, принимает значения [md5, md5-96, sha1, sha1-96, sha2-256, sha2-512, ripemd160].

Необходимый уровень привилегий

15

Значение по умолчанию

Разрешены все алгоритмы аутентификации.

Командный режим

CONFIG

Пример

```
rtt(config)# no ip ssh authentication algorithm md5 disable
```

ip ssh authentication retries

Данная команда устанавливает количество попыток аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает количество попыток аутентификации для SSH-сервера по умолчанию.

Синтаксис

```
ip ssh authentication retries <NUM>
no ip ssh authentication retries
```

Параметры

<NUM> – количество попыток аутентификации для SSH-сервера [1..10].

Необходимый уровень привилегий

10

Значение по умолчанию

6

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh authentication retries 5
```

ip ssh authentication timeout

Данная команда устанавливает, для ssh-сервера на маршрутизаторе, время ожидания ввода пароля при аутентификации SSH-клиента.

Использование отрицательной формы команды (no) устанавливает период времени ожидания аутентификации для SSH-сервера по умолчанию.

Синтаксис

```
ip ssh authentication timeout <SEC>
no ip ssh authentication timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [30..360].

Необходимый уровень привилегий

10

Значение по умолчанию

120

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh authentication timeout 60
```

ip ssh client password

Данной командой определяется пароль по умолчанию для клиента протокола SSH.

Использование отрицательной формы команды (no) удаляет пароль.

Синтаксис

```
ip ssh client password { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no ip ssh client password
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой [1 .. 16] символов, принимает значения [0-9a-fA-F];

<ENCRYPTED-TEXT > – зашифрованный пароль, задаётся строкой [2..32] символов.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh client password test132
```

ip ssh client source-ip

Данной командой определяется IP-адрес маршрутизатора, от которого будут устанавливаться SSH-сессии на другие устройства.

Использование отрицательной формы команды (no) удаляет имя пользователя.

Синтаксис

```
ip ssh client source-ip [ vrf <VRF> ] <ADDR>
no ip ssh client source-ip [ vrf <VRF> ]
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh client source-ip 192.168.22.78
```

ip ssh client username

Данной командой определяется имя пользователя по умолчанию для клиента протокола SSH.

Использование отрицательной формы команды (no) удаляет имя пользователя.

Синтаксис

```
ip ssh client username <NAME>
no ip ssh client username
```

Параметры

<NAME> – имя пользователя, задаётся строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh client username tester
```

ip ssh dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов SSH-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ip ssh dscp <DSCP>
no ip ssh dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

48

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh dscp 40
```

ip ssh encryption algorithm disable

Данная команда запрещает использование определенного алгоритма шифрования для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма шифрования для SSH-сервера.

Синтаксис

```
[no] ip ssh encryption algorithm <ALGORITHM> disable
```

Параметры

<ALGORITHM> – идентификатор алгоритма шифрования, принимает значения [aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, arcfour, arcfour128, arcfour256, blowfish, cast128, 3des].

Необходимый уровень привилегий

15

Значение по умолчанию

Все алгоритмы разрешены.

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh encryption algorithm aes128 disable
```

ip ssh host-key algorithm

Данная команда запрещает использование определенного алгоритма верификации Host-Key для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма алгоритма верификации Host-Key для SSH-сервера.

Синтаксис

```
[no] ip ssh host-key algorithm <ALGORITHM> disable
```

Параметры

<ALGORITHM> – идентификатор алгоритма шифрования, принимает значения [dsa, ecdsa256, ecdsa384, ecdsa521, ed25519, rsa].

Необходимый уровень привилегий

15

Значение по умолчанию

Все алгоритмы разрешены.

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh host-key algorithm dsa disable
```

ip ssh key-exchange algorithm disable

Данная команда запрещает использование определенного алгоритма обмена ключами для SSH-сервера.

Использование отрицательной формы команды (no) разрешает использование определенного алгоритма обмена ключами для SSH-сервера.

Синтаксис

```
[no] ip ssh key-exchange algorithm <ALGORITHM> disable
```

Параметры

<ALGORITHM> – идентификатор протокола обмена ключами, принимает значения [dh-group1-sha1, dh-group14-sha1, dh-group-exchange-sha1, dh-group-exchange-sha256, ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521].

Необходимый уровень привилегий

15

Значение по умолчанию

Все алгоритмы разрешены.

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh key-exchange algorithm dh-group-exchange-sha1 disable
```

ip ssh key-exchange time

Данная команда устанавливает период времени смены ключей аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает период времени смены ключей аутентификации для SSH-сервера по умолчанию.

Синтаксис

```
ip ssh key-exchange time <SEC>
no ip ssh key-exchange time
```

Параметры

<SEC> – период времени в часах, принимает значения [1..72].

Необходимый уровень привилегий

15

Значение по умолчанию

1

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh key-exchange time 24
```

ip ssh key-exchange volume

Данная команда устанавливает объем данных, после прохождения которого произойдет обновление ключей аутентификации для SSH-сервера.

Использование отрицательной формы команды (no) устанавливает объем данных, после прохождения которого произойдет обновление ключей аутентификации для SSH-сервера по умолчанию.

Синтаксис

```
ip ssh key-exchange volume <DATA>
no ip ssh key-exchange volume
```

Параметры

<DATA> – объем данных в мегабайтах, принимает значения [1..4096].

Необходимый уровень привилегий

15

Значение по умолчанию

1000

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh key-exchange volume 512
```

ip ssh port

Данной командой определяется порт SSH-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip ssh port <PORT>
no ip ssh port
```

Параметры

<PORT> – номер порта, указывается в диапазоне [1..65535].

Значение по умолчанию

22

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip ssh port 3001
```

ip ssh server

Данной командой включается SSH-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает SSH-сервер.

Синтаксис

```
[no] ip ssh server [ vrf <VRF>]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать SSH-сервер.

Значение по умолчанию

SSH-сервер выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# no ip ssh server
```

ip telnet access-addresses

Данная команда ограничивает доступ до Telnet-сервера. Telnet-сервер становится доступным только с определённых адресов.

Использование отрицательной формы команды (no) разрешает доступ с любых адресов.

Данная команда позволяет ограничивать доступ до Telnet-сервера при отключенных функциях Firewall или ACL.

В случае совместного использования данного функционала с функциями Firewall и ACL необходимо, чтобы трафик был разрешён всеми тремя функциями.

Синтаксис

```
ip telnet access-addresses [ vrf <NAME> ] <OBJ-GR-NAME>
no ip telnet access-addresses [ vrf <NAME> ]
```

Параметры

<OBJ-GR-NAME> – имя профиля IP-адресов, с которых разрешён доступ.

<VRF> – имя экземпляра VRF, в рамках которого будет работать ограничение доступа на Telnet-сервер.

Необходимый уровень привилегий

15

Значение по умолчанию

Доступ разрешён с любых адресов.

Командный режим

CONFIG

Пример

```
rtt(config)# ip telnet access-addresses MGT
```

ip telnet dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов Telnet-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ip telnet dscp <DSCP>
no ip telnet dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

48

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip telnet dscp 40
```

ip telnet port

Данной командой определяется порт Telnet-сервера на маршрутизаторе.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip telnet port <PORT>
no ip telnet port
```

Параметры

<PORT> – номер порта, принимает значения [1..65535].

Значение по умолчанию

23

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# ip telnet port 2001
```

ip telnet server

Данной командой включается Telnet-сервер на маршрутизаторе.

Использование отрицательной формы команды (no) отключает Telnet-сервер.

Синтаксис

```
[no] ip telnet server [vrf <VRF>]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать Telnet-сервер.

Значение по умолчанию

Telnet-сервер выключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# no ip telnet server
```

show crypto key mypubkey

Команда используется для просмотра открытых ключей устройства, используемых при установлении соединения по протоколу SSH.

Синтаксис

```
show crypto key mypubkey <OPTIONS>
```

Параметры

<OPTIONS> – алгоритм генерации нового криптографического ключа:

- dsa – алгоритм DSA;
- ecdsa – алгоритм ECDSA. Дополнительно необходимо указать размер ключа, 256, 384 или 521;
- ed25519 – алгоритм ED25519;
- rsa – алгоритм RSA;
- rsa1 – алгоритм RSA1.

Необходимый уровень привилегий

15

Командный режим

ROOT

Пример

```
rtt# show crypto key mypubkey rsa
Key data
-----------------------------------------------------------
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDz750sWCQrnNufg1yhuksTFYCYdEfg
JZ9tWUvcssAZhCJWMewprXBuZMABzFmfBg157pgapxn2qJXJ8ESMV7X7gPfy
xQQah6l376z3SFcpKvwudNgwHiS5HCYPRQWx2Xdaz/nJtYr5NpYgLPba68NC
iXcqEp7EPR5GojDVxpuDuk0hPFcihzmt5Yx8ZptJRzRtsuDQYlowv0Qa24kd
OlQ90/1qKfbAhB6XI60l+dK5VEj7giBESarcRn69/e/YVbdGBdTE93QWFPKI
bm63imfbxRwWtcwsFdIHi8Blv9ZqDqqF/IO3TkIKa31hV9GnsawlAXi/IdyY
bYPboHRdcTlH/ root@rtt
```

ssh authentication method

Данной командой выбирается метод аутентификации SSH-сессий для выбранной учетной записи.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ssh authentication method <METHOD>
no ssh authentication method
```

Параметры

<METHOD> – метод аутентификации SSH-сессий. Может принимать значения:

- password – аутентификация пользователя при открытии SSH-сессий может быть произведена только по паролю;
- pubkey – аутентификация пользователя при открытии SSH-сессий может быть произведена только по публичному ключу;
- both – аутентификация пользователя при открытии SSH-сессий может быть произведена как по паролю, так и по публичному ключу.

Значение по умолчанию

Аутентификация пользователя при открытии SSH-сессии может быть произведена только по паролю.

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

```
rtt(config-user)# ssh authentication method pubkey
```

ssh pubkey

Данной командой указывается публичный ключ, который будет использован при аутентификации SSH-сессии для выбранной учетной записи.

Использование отрицательной формы команды (no) удаляет привязку публичного ключа к учетной записи из конфигурации.

Синтаксис

```
ssh pubkey <NAME>
no ssh pubkey
```

Параметры

<NAME> – имя файла публичного ключа, расположенного в разделе crypto:public-key, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует

Необходимый уровень привилегий

15

Командный режим

CONFIG-USER

Пример

```
rtt(config-user)# ssh pubkey admin_id_rsa.pub
```

update ssh-host-key

Данной командой генерируется пара новых криптографических ключей для установления соединения по протоколу SSH.

Синтаксис

```
update ssh-host-key { dsa | escda <ESCDA> | ed25519 <ED25519> | rsa <RSA> }
```

Параметры

dsa – алгоритм DSA;

ecdsa – алгоритм ECDSA:

- <ECDSA> – размер ключа, принимает значение 256, 384 или 521;
- Без указания используется размер ключа 521.

ed25519 – алгоритм ED25519:

- <ED25519> – размер ключа, принимает значение [256..2048];
- Без указания используется размер ключа 2048.

rsa – алгоритм RSA с указанием длины ключа:

- <RSA> – размер ключа, принимает значение [1024..2048];
- Без указания используется размер ключа 2048.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# update ssh-host-key ecdsa
```

27.6. Настройка протокола LLDP

lldp enable

Данной командой на маршрутизаторе включается поддержка протокола LLDP.

Использование отрицательной формы команды (no) отключает поддержку протокола LLDP.

Синтаксис

```
[no] lldp enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# lldp enable
```

lldp hold-multiplier

Данной командой устанавливается количество периодов lldp-timer, в течение которых маршрутизатор хранит информацию, полученную по протоколу LLDP.

Использование отрицательной формы команды (no) устанавливает соответствующий параметр в значение по умолчанию.

Синтаксис

```
lldp hold-multiplier <SEC>
no lldp hold-multiplier
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..10].

Значение по умолчанию

4

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# lldp hold-multiplier 5
```

lldp management-address

Данной командой устанавливается IP-адрес, который будет передаваться в LLDP TLV в качестве management-address.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию для поля LLDP TLV management-address.

Синтаксис

```
lldp management-address <ADDR>
no lldp management-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Указываемый IP-адрес должен быть назначен на каком-либо интерфейсе/туннеле маршрутизатора.

Значение по умолчанию

Один из существующих.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# lldp management-address 192.168.54.42
```

lldp med fast-start enable

Данной командой включается рассылка LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой (см. раздел lldp network-policy).

Использование отрицательной формы команды (no) выключает рассылку LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой.

Синтаксис

```
[no] lldp med fast-start enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Рассылка LLDP-MED сообщений отключена.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# lldp med fast-start enable
```

lldp med fast-start tx-interval

Данной командой настраивается период рассылки LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой (см. раздел lldp network-policy).

Использование отрицательной формы команды (no) устанавливается значение по умолчанию периода рассылки LLDP-MED сообщений на интерфейсах с настроенной сетевой политикой.

Синтаксис

```
lldp med fast-start tx-interval <SEC>
[no] lldp med fast-start tx-interval
```

Параметры

<SEC> – период времени, принимает значения [1.. 32768] секунд.

Значение по умолчанию

1 секунда.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# lldp med fast-start tx-interval 20
```

lldp receive

Данной командой включается обработка LLDPDU на физическом интерфейсе.

Использование отрицательной формы команды (no) отключает обработку LLDPDU на физическом интерфейсе.

Синтаксис

```
[no] lldp receive
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Прием LLDP-пакетов выключен.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

Пример

```
rtt(config-if-gi)# lldp receive
```

lldp system-description

Данной командой устанавливается поле, который будет передаваться в LLDP TLV в качестве system-description.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию для поля LLDP TLV system-description.

Синтаксис

```
lldp system-description <DESCRIPTION>
no lldp system-description
```

Параметры

<DESCRIPTION> – идентификатор описания системы, задаётся строкой до 255 символа.

Значение по умолчанию

Содержит информацию о модели и версии ПО маршрутизатора.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# lldp system-description rtt-1.3.0
```

lldp system-name

Данной командой устанавливается поле, который будет передаваться в LLDP TLV в качестве system-name.

Использование отрицательной формы команды (no).

Синтаксис

```
lldp system-name <NAME>
no lldp system-name
```

Параметры

<NAME> – идентификатор имени системы, задаётся строкой до 255 символа.

Значение по умолчанию

Совпадает с заданным hostname.

Необходимый уровень привилегий

15

Командный режим

CONFIG

Пример

```
rtt(config)# lldp system-name rtt-branch-12
```

lldp timer

Данной командой устанавливается период отправки LLDPDU.

Использование отрицательной формы команды (no) устанавливает период отправки LLDPDU по умолчанию.

Синтаксис

```
lldp timer <SEC>
no lldp timer
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..32768].

Значение по умолчанию

30

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# lldp timer 3
```

lldp transmit

Данной командой включается отправка LLDPDU на физическом интерфейсе.

Использование отрицательной формы команды (no), отправка LLDPDU на физическом интерфейсе отключается.

Синтаксис

```
[no] lldp transmit
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отправка LLDP-пакетов выключена.

Необходимый уровень привилегий

15

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

Пример

```
rtt(config-if-gi)# lldp transmit
```

show lldp neighbors

Данной командой выводится информация о подключенных устройствах, от которых получена информация по протоколу LLDP.

Синтаксис

```
show lldp neighbors [ <IF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<IF> – наименование физического интерфейса или группы физических интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.
Допустимо использовать только физические интерфейсы (gigabitethernet или tengigabitethernet). Без указания интерфейса выводится информация о всех обнаруженных по LLDP устройствах.

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения информации по протоколу LLDP в формате вывода без включенного кластера.

При включенном кластере отображается информация для всех юнитов кластера.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# show lldp neighbors
LLDP Neighbor Information:

Local Information:
    Index:              0
    Local Interface:    gi1/0/4

Neighbour Information:
    Chassis type:       mac
    Chassis ID:         a8:f9:4b:aa:8c:90
    Management ip:      192.168.1.5
    Management ip:      fe80::2052:e5ff:fe36:226f
    Port type:          local
    Port ID:            gi1/0/8
    Port description:   rtt200-lldp-test
    Time to live:       120
    System name:        rtt-test
    System Description: Rusteletech Router R200 1.3.0 build 79 (date 14/08/2017 time
10:19:13)

System capabilities:
    Bridge:             false
    Router:             true
    Station:            true
    Wlan:               false
```

show lldp statistics

Данной командой выводится статистика работы протокола LLDP на интерфейсах, на которых включен протокол LLDP.

Синтаксис

```
show lldp statistics [ <IF> ] [ unit { <RANGE-ID> | local } ]
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора. Допустимо использовать только физические интерфейсы (gigabitethernet или tengigabitethernet). Без указания интерфейса выводится информация о всех обнаруженных по LLDP устройствах.

<RANGE-ID> – принимает номера юнитов в формате: ID,ID-ID. Номер юнита принимает значения [1..4];

local – ключ для отображения статистики работы протокола LLDP в формате вывода без включенного кластера.

При включенном кластере отображается информация для всех юнитов кластера.show lldp statistics

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show lldp statistics
Interface  Transmitted  Received  Discarded  Unrecognized  Ageout  Inserted   Deleted
---------  -----------  --------  ---------  ------------  ------  --------   -------
 gi1/0/1         1           0          0            0          0         0          0
```

27.7. Настройка зеркалирования

port monitor

Данной командой включается режим зеркалирования трафика.

Использование отрицательной формы команды (no) отключается режим зеркалирования трафика.

Синтаксис

```
[no] port monitor
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

port monitor interface

Данной командой определяются контролируемые порты.

Использование отрицательной формы команды (no) удаляет контролируемый порт.

Синтаксис

```
port monitor interface <IF> <DIRECTION>
no port monitor
```

Параметры

<IF> – физический интерфейс или группа физических интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<DIRECTION> – направление трафика:

- tx – зеркалирование только исходящего трафика;
- rx – зеркалирование только входящего трафика.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

Пример

```
rtt(config-if-gi)# port monitor interface gigabitethernet 1/0/5
```

port monitor mode

Данной командой определяется режим порта, передающего отзеркалированный трафик.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
port monitor mode <MODE>
no port monitor mode
```

Параметры

<MODE> – режим:

- network – совмещенный режим передачи данных и зеркалирование;
- monitor-only – только зеркалирование.

Значение по умолчанию

network

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# port monitor mode monitor-only
```

port monitor remote

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой включает режим удаленного зеркалирования (RSPAN).

Использование отрицательной формы команды (no) отключает удаленное зеркалирование (RSPAN).

Синтаксис

```
[no] port monitor remote
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

Пример

```
rtt(config-if-gi)# port monitor remote
```

port monitor remote vlan

В текущей версии ПО данная команда поддерживается только на маршрутизаторах R800.

Данной командой определяется VLAN, по которому будет передаваться отзеркалированный трафик.

Использование отрицательной формы команды (no) удаляет указанный VLAN.

Синтаксис

```
port monitor remote vlan <VID> <DIRECTION>
no port monitor remote vlan <DIRECTION>
```

Параметры

<VID> – идентификационный номер VLAN, задаётся в диапазоне [2…4094];

<DIRECTION> – направление трафика:

- tx – зеркалирование в указанный VLAN только исходящего трафика;
- rx – зеркалирование в указанный VLAN только входящего трафика.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# port monitor remote vlan 10
```

show interfaces switch-port monitor

Команда используется для просмотра настроек зеркалирования.

Синтаксис

```
show interfaces switch-port monitor
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces switch-port monitor
Port monitor mode:         network
RSPAN configuration RX:    VLAN 222
RSPAN configuration TX:    VLAN 222
Source Port    Destination Port   Type     RSPAN
------------   ----------------   ------   --------
gi1/0/7        gi1/0/6            RX,TX    Enabled
```

27.8. Настройка zabbix-agent и zabbix-proxy

active-server

Данной командой задается адрес и порт Zabbix-сервера для активных проверок.

Использование отрицательной формы команды (no) удаляет сервер для активных проверок.

Синтаксис

```
active-server <ADDR> [ port <PORT> ]
no active-server
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – номер порта, указывается в диапазоне [1..65535].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

Пример

```
rtt(config-zabbix-agent)# active-server 192.168.16.54
```

config-retrieve

Данной командой задаётся интервал обновления данных конфигурации от сервера.

Использование отрицательной формы команды (no) восстанавливает значение по умолчанию.

Синтаксис

```
config-retrieve <SEC>
no config-retrieve
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..604800].

Значение по умолчанию

60

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-proxy)# config-retrieve 200
```

database

Данной командой устанавливается место хранения базы данных zabbix-proxy.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
database <PATH>
no database
```

Параметры

<PATH> – место хранения базы данных zabbix-proxy.

Значение по умолчанию

По умолчанию база данных zabbix хранится в энергозависимой памяти маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# database flash:data/zabbix.dat
```

enable

Данной командой активируется функционал Zabbix-агента.

Использование отрицательной формы команды (no) отключает функционал Zabbix-агента.

Синтаксис

```
[no] enable
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# enable
```

hostname

Данной командой задается имя агента. Hostname должен совпадать с именем хоста в панели управления Zabbix.

Использование отрицательной формы команды (no) устанавливает имя агента по умолчанию.

Синтаксис

```
hostname <NAME>
no hostname
```

Параметры

<NAME> – идентификатор имени системы, задаётся строкой до 63 символов.

Значение по умолчанию

Совпадает с настроенным hostname маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# hostname branch_12
```

port

Данной командой задается TCP-порт, который будет использоваться для обращения Zabbix-сервера/прокси к агенту.

Использование отрицательной формы команды (no) удаляет значение TCP-порта.

Синтаксис

```
port <PORT>
[no] port
```

Параметры

<PORT> – номер порта, указывается в диапазоне [1024..32767].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# port 20050
```

remote-commands

Данной командой активируем возможность выполнение команд на маршрутизаторе с Zabbix-сервера.

Использование отрицательной формы команды (no) отключает возможность выполнение команд на маршрутизаторе с Zabbix-сервера.

Синтаксис

```
[no] remote-commands
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

15

Командный режим

CONFIG-ZABBIX-AGENT

Пример

```
rtt(config-zabbix-agent)# remote-commands
```

server

Данной командой задается IP-адрес Zabbix-сервера для zabbix-agent, с которого разрешено принимать входящие соединения.

Использование отрицательной формы команды (no) удаляет IP-адрес Zabbix-сервера.

Синтаксис

```
server <ADDR>
no server
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

Пример

```
rtt(config-zabbix-agent)# server 192.168.18.54
```

server

Данной командой задается IP-адрес и порт Zabbix-сервера для zabbix-proxy, с которого разрешено принимать входящие соединения.

Использование отрицательной формы команды (no) удаляет конфигурацию Zabbix-сервера для zabbix-proxy.

Синтаксис

```
server <ADDR> [port <PORT>]
no server
```

Параметры

<ADDR> – IP-адрес сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<PORT> – номер UDP-порта, указывается в диапазоне [1024..32767].

Значение по умолчанию

<ADDR> – отсутствует;

<PORT> – 10051.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# server 192.168.18.54 port 1285
```

show zabbix-agent configuration

Данной командой осуществляется вывод конфигурации Zabbix-агента.

Синтаксис

```
show zabbix-agent configuration
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show zabbix-agent configuration
VRF:                          --
State:                        Enabled
Active server:                --
Active server port:           --
Hostname:                     rtt
Port:                         --
Remote commands:              Disabled
Server:                       192.168.18.54
Source address:               --
Timeout:                      3
```

show zabbix-proxy configuration

Данной командой осуществляется вывод конфигурации Zabbix-proxy.

Синтаксис

show zabbix-proxy configuration

Параметры

Отсутствуют.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show zabbix-proxy configuration
VRF:                          --
State:                        Enabled
Hostname:                     RTT-SLA
```

```
Port:                          --
Remote commands:               Enabled
Server:                        192.168.32.228
Server port:                   --
Source address:                --
Timeout:                       3
Configuration retrieve:        30
```

source-address

Данной командой указывается адрес с которого будет устанавливаться соединение с Zabbix-сервером.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
source-address <ADDR>
no source-address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Значение по умолчанию

IP-адрес интерфейса, с которого отправляются пакеты на Zabbix-сервер.

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# source-address 192.168.1.57
```

timeout

Данной командой задается максимальное время на обработку запроса Zabbix-сервера.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
timeout
no timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..30].

Значение по умолчанию

3

Необходимый уровень привилегий

10

Командный режим

CONFIG-ZABBIX-AGENT

CONFIG-ZABBIX-PROXY

Пример

```
rtt(config-zabbix-agent)# timeout 20
```

zabbix-agent

Данной командой осуществляется переход в режим конфигурирования Zabbix-агента в глобальном режиме или VRF.

Использование отрицательной формы команды (no) удаляет настройки Zabbix-агента в глобальном режиме или VRF.

Синтаксис

```
[no] zabbix-agent [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

Значение по умолчанию

Отключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# zabbix-agent
```

zabbix-proxy

Данной командой осуществляется переход в режим конфигурирования Zabbix-proxy в глобальном режиме или VRF.

Использование отрицательной формы команды (no) удаляет настройки Zabbix-proxy в глобальном режиме или VRF.

Синтаксис

```
[no] zabbix-proxy [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# zabbix-proxy
```

28. НАСТРОЙКА DHCP

28.1. Управление DHCP-клиентом

ip address dhcp

Данной командой включается получение динамического IP-адреса конфигурируемого интерфейса по протоколу DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IP-адреса по протоколу DHCP.

Синтаксис

```
[no] ip address dhcp
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip address dhcp
```

ip dhcp client ignore

Данной командой указываются DHCP-опции, которые будут игнорироваться клиентом.

Синтаксис

```
[no] ip dhcp client ignore <OPTION>
```

Параметры

<OPTION> – принимает следующие значения:

- nameserver – DHCP-опция 23, список DNS-серверов;
- domain-name – DHCP-опция 24, имя домена;
- netbios-nameserver – DHCP-опция 44, список NetBios-серверов;
- router – DHCP-опция 3, список шлюзов по умолчанию;
- classless-static-route – DHCP-опция 121, список бесклассовых статических маршрутов;
- classful-static-route – DHCP-опция 33, список классовых статических маршрутов;
- tftp-server-address – DHCP-опция 66, имя TFTP-сервера (не применима для Bridge-интерфейсов);
- vendor-specific – DHCP-опция 17, информация определенная производителем.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client ignore router
```

ip dhcp client lease-time

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp client lease-time <TIME>
no ip dhcp client lease-time
```

Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

- DD – дни, принимает значение [0..364];
- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59].

Значение по умолчанию

1 день.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client lease-time 00:12:00
```

ip dhcp client reboot

Данной командой задаётся время, в течение которого DHCP-клиент будет пытаться получить старый IP-адрес, перед тем как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp client reboot <SEC>
no ip dhcp client reboot
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client reboot 60
```

ip dhcp client retry

Данной командой задаётся интервал, через который DHCP-клиент возобновит попытки получить IP-адрес, если было установлено, что DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp client retry <SEC>
no ip dhcp client retry
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

Значение по умолчанию

300

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client retry 180
```

ip dhcp client select-timeout

Данной командой задаётся период времени, в течение которого DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими DHCP-серверами. Клиенту в ответ на запрос IP-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждет указанный период времени с момента отправки запроса на получение IP-адреса на случай, если он получит несколько предложений от различных DHCP-серверов. По истечении указанного времени клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp client select-timeout <SEC>
no ip dhcp client select-timeout
```

Параметры

<SEC> – период времени в секундах, принимает значение [0..600].

Значение по умолчанию

0 секунд – клиент примет первое полученное предложение.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client select-timeout 30
```

ip dhcp client timeout

Данной командой задаётся интервал, по истечении которого клиент считает, что DHCP-сервер недоступен. Если в базе данных IP-адресов клиента есть какие-либо арендованные адреса, срок аренды которых еще не истек, то клиент будет проверять последовательно каждый из них и, если найдет корректную, то IP-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IP-адрес по истечении интервала повтора (dhcp retry). Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp client timeout <SEC>
no ip dhcp client timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1 .. 600].

Значение по умолчанию

60 секунд.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client timeout 300
```

ip dhcp client vendor-class-id

Данной командой устанавливается значение DHCP Опции 60 для получения дополнительных настроек по DHCP Опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

Синтаксис

```
ip dhcp client vendor-class-id <NAME>
no ip dhcp client vendor-class-id
```

Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp client vendor-class-id Rusteletech
```

ip dhcp server address

Данной командой устанавливается IP-адрес DHCP-сервера, с которого будет получен IP-адрес (и другие опции). Предложения от других DHCP-серверов будут проигнорированы.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес DHCP-сервера.

Синтаксис

```
ip dhcp server address <ADDR>
no ip dhcp server address
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-OOB

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip dhcp server address 10.10.10.1
```

renew dhcp

Данной командой отправляется запрос на обновление IP-адреса на по протоколу DHCP.

Синтаксис

```
renew dhcp <IF>
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# renew dhcp gigabitethernet 1/0/1
```

28.2. Управление IPv6 DHCP-клиентом

ipv6 address dhcp

Данной командой включается получение динамического IPv6-адреса конфигурируемого интерфейса по протоколу IPv6 DHCP.

Использование отрицательной формы команды (no) выключает получение динамического IPv6-адреса по протоколу IPv6 DHCP.

Синтаксис

```
[no] ipv6 address dhcp
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 address dhcp
```

ipv6 dhcp client ignore

Данной командой указываются DHCP-опции, которые будут игнорироваться клиентом.

Синтаксис

```
ipv6 dhcp client ignore <OPTION>
no ipv6 dhcp client ignore
```

Параметры

<OPTION> – принимает следующие значения:

- dns-nameserver – DHCP-опция 23, список DNS-серверов;
- domain-name – DHCP-опция 24, имя домена;
- vendor-specific – DHCP-опция 17, информация, определенная производителем.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client ignore dns-nameserver
```

ipv6 dhcp client lease-time

Данной командой устанавливается запрашиваемое время аренды сетевого адреса.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 dhcp client lease-time <TIME>
no ipv6 dhcp client lease-time
```

Параметры

<TIME> – запрашиваемое время аренды, задаётся в виде DD:HH:MM, где:

- DD – дни, принимает значение [0..364];
- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59].

Значение по умолчанию

1 день.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client lease-time 00:12:00
```

ipv6 dhcp client reboot

Данной командой задаётся время, в течение которого IPv6 DHCP-клиент будет пытаться получить старый IPv6-адрес перед тем, как начать получать новый.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 dhcp client reboot <SEC>
no ipv6 dhcp client reboot
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client reboot 60
```

ipv6 dhcp client retry

Данной командой задаётся интервал, через который IPv6 DHCP-клиент возобновит попытки получить IPv6-адрес, если было установлено, что IPv6 DHCP-сервер не отвечает.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 dhcp client retry <SEC>
no ipv6 dhcp client retry
```

Параметры

<SEC> – период времени в секундах, принимает значение [1..600].

Значение по умолчанию

300

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client retry 180
```

ipv6 dhcp client select-timeout

Данной командой задаётся период времени, в течение которого IPv6 DHCP-клиент будет выбирать среди предложений по аренде от серверов, если такие существуют. Это используется в сетях с несколькими IPv6 DHCP-серверами – в этом случае клиенту в ответ на запрос IPv6-адреса может быть отправлено несколько предложений. Возможно, что одно из этих предложений предпочтительнее другого (например, одно предложение может иметь адрес, который клиент использовал ранее).

Клиент ждет указанный период времени с момента отправки запроса на получение IP-адреса на случай, если он получит несколько предложений от различных DHCPv6-серверов. По истечении указанного времени клиент принимает одно из предложений.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 dhcp client select-timeout <SEC>
no ipv6 dhcp client select-timeout
```

Параметры

<SEC> – период времени в секундах, принимает значение [0..600].

Значение по умолчанию

0 – клиент примет первое предложенное предложение.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client select-timeout 30
```

ipv6 dhcp client timeout

Данной командой задаётся интервал, по истечении которого клиент считает, что IPv6 DHCP-сервер недоступен. Если в базе данных IPv6-адресов клиента есть какие-либо арендованные адреса, срок аренды которых ещё не истек, то клиент будет проверять последовательно каждый из них, и если найдет корректную, то IPv6-адрес из неё будет присвоен интерфейсу. Если нет действующих аренд в базе данных, то клиент будет повторно запрашивать IPv6-адрес по истечении интервала повтора (dhcp retry).

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ipv6 dhcp client timeout <SEC>
no ipv6 dhcp client timeout
```

Параметры

<SEC> – период времени в секундах, принимает значения [1..600].

Значение по умолчанию

60 секунд.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client timeout 300
```

ipv6 dhcp client vendor-class-id

Данной командой устанавливается значение DHCP опции 60 для получения дополнительных настроек по DHCP опции 43.

Использование отрицательной формы команды (no) отключает запрос данной опции.

Синтаксис

```
ipv6 dhcp client vendor-class-id <NAME>
no ipv6 dhcp client vendor-class-id
```

Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp client vendor-class-id Rusteletech
```

ipv6 dhcp server address

Данной командой устанавливается IPv6-адрес DHCP-сервера, с которого будет получен IPv6-адрес (и другие опции). Предложения от других DHCP-серверов будут проигнорированы.

Использование отрицательной формы команды (no) удаляет установленный IPv6-адрес DHCP-сервера.

Синтаксис

```
ipv6 dhcp server address <IPV6-ADDR>
no ipv6 dhcp server address
```

Параметры

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp server address fc00::1
```

renew ipv6 dhcp

Данной командой отправляется запрос на обновление IPv6-адреса на по протоколу DHCP.

Синтаксис

```
renew ipv6 dhcp <IF>
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# renew ipv6 dhcp gigabitethernet 1/0/1
```

28.3. Управление DHCP Relay агентом

ip dhcp information option

Данной командой включается обработка опции 82 DHCP Relay агентом.

Использование отрицательной формы команды (no) отключает обработку опции 82 DHCP Relay агентом.

Синтаксис

```
[no] ip dhcp information option
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp information option
```

ip dhcp information option-insert

Данной командой включается вставка DHCP Relay агентом, работающим на multipoint GRE туннеле RTT в роли NHRP NHS, 82 опции в DHCP-запросы от NHRP NHC. В добавляемой опции указывается NBMA-адрес NHRP NHC.

Использование отрицательной формы команды (no) отключает вставка опции 82 DHCP Relay агентом.

Синтаксис

```
[no] ip dhcp information option-insert
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-GRE

Пример

```
rtt(config-gre)# ip dhcp information option-insert
```

ip dhcp information option action

Данная команда позволяет изменить решение, которое будет принято маршрутизатором при получении пакета с установленной опцией 82.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
ip dhcp information option action { replace | keep | drop }
no ip dhcp information option action
```

Параметры

- replace – заменить установленную опцию 82 новой;
- keep – оставить установленную опцию 82 без изменений;
- drop – удалить пакет, если в нем обнаружена установленная опция 82.

Значение по умолчанию

keep

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip dhcp information option action drop
```

ip dhcp information option format-type access-node-id

Данная команда позволяет задать Access-node-ID в поле Circuit ID опции 82 при использовании формата tr101.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip dhcp information option format-type access-node-id <NODE-ID>
```

Параметры

<NODE-ID> – значение Access-node-ID, задаётся строкой до 31 символа.

Значение по умолчанию

Hostname устройства.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp information option format-type access-node-id R1
```

ip dhcp information option format-type circuit-id

Данная команда позволяет задать значение поля Circuit ID опции 82 при использовании любого формата.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip dhcp information option format-type circuit-id <CIRCUIT-ID>
```

Параметры

<CIRCUIT-ID> – значение Circuit ID, задаётся строкой до 63 символов.

Значение по умолчанию

Определяется используемым форматом.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip dhcp information option format-type circuit-id gi1_0_1
```

ip dhcp information option format-type option

Данная команда позволяет задать формат идентификатора порта в поле Circuit ID опции 82 при использовании формата tr101.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip dhcp information option format-type option <FORMAT> [ delimeter <DELIMITER>
]
```

Параметры

<FORMAT> – формат идентификатора порта, принимает одно из следующих значений:

- sp – строка, содержащая номер слота и порта;
- sv – строка, содержащая номер слота и VLAN;
- pv – строка, содержащая номер порта и VLAN;
- spv – строка, содержащая номер слота, порта и VLAN;
- bin – бинарный формат: VLAN (4 байта), слот (1 байт), порт (4 байта).

<DELIMITER> – разделитель между параметрами в строке, может принимать одно из следующих значений [. , ; # / space].

Значение по умолчанию

Формат – spv

Разделитель «:» для форматов: sp, sv, pv

Разделитель «/» и «:» для формата spv

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp information option format-type option spv delimeter #
```

ip dhcp information option format-type remote-id

Данная команда позволяет задать значение поля Remote ID опции 82 при использовании любого формата.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] ip dhcp information option format-type remote-id <REMOTE-ID>
```

Параметры

<REMOTE-ID> – значение Remote ID, задаётся строкой до 63 символов.

Значение по умолчанию

Определяется используемым форматом.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip dhcp information option format-type remote-id R1
```

ip dhcp information option suboption-type

Данной командой выбирается формат опции 82.

Использование отрицательной формы команды (no) устанавливает формат по умолчанию.

Синтаксис

```
[no] ip dhcp information option suboption-type { tr101 | custom }
```

Параметры

tr101 – формат, рекомендованный TR-101 опции 82 согласно синтаксису, принятому в рекомендациях TR-101, приведен в таблице 10;

custom – формат опции 82 в соответствии с форматом, приведенном в таблице 11.

Таблица 10 – Формат полей опции 82 согласно рекомендациям TR-101

Поле	Передаваемая информация
Circuit ID	Access-Node-Id, по умолчанию hostname устройства (формат можно изменить командой «ip dhcp information option format-type access-node-id», описанной в разделе ip dhcp information option format-type access-node-id) Строка вида eth <stacked/slotid/interfaceid>:<vlan> (формат можно изменить командой «ip dhcp information option format-type option», описанной в разделе ip dhcp information option format-type option) Последний байт – номер порта, к которому подключено устройство, отправляющее dhcp-запрос.
Remote agent ID	Enterprise number – 0089c1 MAC-адрес устройства

Таблица 11 – Формат полей опции 82 режима custom

Поле	Передаваемая информация
Circuit ID	Длина (1 байт) Тип Circuit ID Длина (1 байт) VLAN (2 байта) Номер модуля (1 байт) Номер порта (4 байта)
Remote agent ID	Длина (1 байт) Тип Remote ID (1 байт) Длина (1 байт) MAC-адрес коммутатора

Значение по умолчанию

tr101

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp information option suboption-type custom
```

ip dhcp-relay

Данная команда включает агент DHCP-relay.

Использование отрицательной формы команды (no) отключает агент DHCP-relay.

Синтаксис

```
[no] ip dhcp-relay
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-relay
```

ip helper-address

Данной командой указывается IP-адрес DHCP-сервера, которому будут отправляться DHCP Discover пакеты, перехваченные DHCP Relay-агентом.

Использование отрицательной формы команды (no) удаляет IP-адрес из списка DHCP-серверов для DHCP Relay-агента.

Синтаксис

```
[no] ip helper-address <IP>
```

Параметры

<IP> – IP-адрес DHCP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 6 IP-адресов, список задаётся через запятую.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

Пример

```
rtt(config-if-gi)# ip helper-address 10.10.10.1
```

ip helper-address gateway-ip

Данной командой выставляется необходимый IP-адрес в поле Gateway-IP в DHCP Offer пакетах, отправляемых DHCP Relay-агентом.

Использование отрицательной формы команды (no) выставляет ip-address в поле Gateway-IP по умолчанию.

Синтаксис

```
ip helper-address gateway-ip <ADDR>
no ip helper-address gateway-ip
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip helper-address gateway-ip 10.10.10.1
```

ip helper-address vrrp-group

Данной командой запрещается отправка DHCP Discover пакетов, перехваченные DHCP Relay-агентом, если VRRP-группа находится в состоянии DOWN.

Использование отрицательной формы команды (no) разрешает отправку DHCP Discover пакетов, перехваченные DHCP Relay-агентом, если VRRP-группа находится в состоянии DOWN.

Синтаксис

```
ip helper-address vrrp-group <GRID>
no ip helper-address vrrp-group
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ip helper-address vrrp-group 7
```

28.4. Управление IPv6 DHCP Relay агентом

ipv6 dhcp-relay

Данная команда включает IPv6 DHCP-агент.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-агент.

Синтаксис

```
[no] ipv6 dhcp-relay
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 dhcp-relay
```

ipv6 dhcp-relay interface

Данной командой указывается исходящий интерфейс и IPv6-адрес DHCP-сервера, которому будут отправляться DHCP Solicit-пакеты, перехваченные IPv6 DHCP Relay-агентом.

Использование отрицательной формы команды (no) удаляет исходящий интерфейс из списка интерфейсов для IPv6 DHCP Relay-агента.

Синтаксис

```
ipv6 dhcp-relay interface <IF> [ <IPV6-ADDR> ]
no ipv6 dhcp-relay interface <IF>
```

Параметры

<IF> – наименование интерфейса или группы интерфейсов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

<IPV6-ADDR> – IPv6-адрес DHCP-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если адрес не задан, то пакеты отправляются на ff02::1:2. Все IPv6 DHCP-сервера и агенты в локальном сетевом сегменте.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

Пример

```
rtt(config-if-gi)# ipv6 dhcp-relay interface gigabitethernet 1/0/2 fc00::1
```

28.5. Настройка и мониторинг DHCP-сервера

address

Данная команда позволяет добавить IP-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес.

Синтаксис

```
address <ADDR> { mac-address <MAC> | client-identifier <CLIENT-ID> } [DESCRIPTION]
no address <ADDR>
```

Параметры

<ADDR> – IP-адрес клиента, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IP-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IP-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

<CLIENT-ID> – идентификатор клиента согласно DHCP option 61;

<DESCRIPTION> – описание заданного IP-адреса, задаётся строкой до 255 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# address 192.168.3.21 mac-address A8:F9:4B:AA:00:40
```

address-range

Данная команда позволяет добавить диапазон IP-адресов к пулу адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IP-адресов.

Синтаксис

```
address-range <FROM-ADDR>-<TO-ADDR>
no address-range { <FROM-ADDR>-<TO-ADDR> | all }
```

Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IP-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# address-range 192.168.3.1-192.168.3.20,192.168.3.24
```

clear ip dhcp binding

Данная команда позволяет очистить таблицу выданные DHCP-сервером IP-адресов.

Синтаксис

```
clear ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

Параметры

<ADDR> – IP-адрес, опциональный параметр, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если в команде задан параметр <ADDR>, то будет отображена информация связанная только с указанным адресом.

<VRF> – имя экземпляра VRF, по которому будет выводиться информация. Задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

ROOT

Пример

```
rtt# clear ip dhcp binding
```

default-lease-time

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IP-адрес, если клиент не запрашивал определенное время аренды. Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
default-lease-time <TIME>
no default-lease-time
```

Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

Значение по умолчанию

12 часов

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# default-lease-time 00:04:00
```

default-router

Данная команда позволяет задать список IP-адресов шлюзов по умолчанию, которые DHCP-сервер будет сообщать клиентам, используя DHCP опцию 3.

Использование отрицательной формы команды (no) удаляет указанные адреса из списка шлюзов.

Синтаксис

```
default-router <ADDR>
no default-router { <ADDR> | all }
```

Параметры

<ADDR> – IP-адрес шлюза по умолчанию, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все шлюзы по умолчанию.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# default-router 192.168.3.1,192.168.3.2
```

dns-server

Данная команда позволяет задать список IP-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

Синтаксис

```
dns-server <ADDR>
no dns-server { <ADDR> | all }
```

Параметры

<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 8 IP-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

domain-name

Данная команда позволяет задать DNS-имя сетевого домена. Имя домена передается клиентам в составе DHCP-опций 15.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

Синтаксис

```
domain-name <NAME>
no domain-name
```

Параметры

<NAME> – DNS-имя домена клиента, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# domain-name Rusteletech.loc
```

excluded-address-range

Данная команда позволяет исключить диапазон IP-адресов из пула адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IP-адресов.

Синтаксис

```
[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>
```

Параметры

<FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# excluded-address-range 192.168.3.1-
192.168.3.20,192.168.3.24
```

ip dhcp-server

Данная команда включает DHCP-сервер.

Использование отрицательной формы команды (no) выключает DHCP-сервер.

Синтаксис

```
[no] ip dhcp-server [vrf <VRF>]
```

Параметры

<VRF> – имя экземпляра VRF, в рамках которого будет работать DHCP-сервер. Задается строкой до 31 символа.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-server
```

ip dhcp-server dscp

Команда задаёт значение кода DSCP для использования в IP-заголовке исходящих пакетов DHCP-сервера.

Использование отрицательной формы команды (no) устанавливает значение DSCP по умолчанию.

Синтаксис

```
ip dhcp-server dscp <DSCP>
no ip dhcp-server dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].

Значение по умолчанию

61

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-server dscp 40
```

ip dhcp-server pool

Команда используется для создания пула IP-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IP-адресов.

Синтаксис

```
ip dhcp-server pool <NAME> [vrf <VRF>]
no ip dhcp-server pool { <NAME> | all }
```

Параметры

<NAME> – имя пула IP-адресов DHCP-сервера, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IP-адресов. Если использовать команду для удаления, то при указании значения «all» будут удалены все IP-адреса;

<VRF> – имя экземпляра VRF, в рамках которого будет работать данный пул IP-адресов DHCP-сервера. Задается строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-server pool lan
```

ip dhcp-server vendor-class-id

Данная команда необходима для создания идентификатора класса поставщика (DHCP опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

Синтаксис

```
[no] ip dhcp-server vendor-class-id <NAME>
```

Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip dhcp-server vendor-class-id Rusteletech
```

max-lease-time

Данная команда позволяет задать максимальное время аренды IP-адресов. Если DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
max-lease-time <TIME>
no max-lease-time
```

Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

Значение по умолчанию

1 день.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# max-lease-time 00:16:00
```

netbios-nameserver

Данная команда позволяет сконфигурировать 44 опцию DHCP (задает IP-адрес NetBIOS-сервера).

Использование отрицательной формы команды (no) выключает передачу IP-адреса NetBIOS-сервера (44 опцию).

Синтаксис

```
[no] netbios-nameserver <ADDR>
```

Параметры

<ADDR> – IP-адрес NetBIOS-сервера задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно задать до 4 IP-адресов. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все IP-адреса.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# netbios-nameserver 192.168.45.15
```

network

Данная команда задает IP-адрес и маску для подсети, из которой будет выделен пул IP-адресов.

Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

Синтаксис

```
network <ADDR/LEN>
```

```
no network
```

Параметры

<ADDR/LEN> – IP-подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# network 192.168.3.0/24
```

next-server

Данная команда позволяет сконфигурировать 66 опцию DHCP.

Использование отрицательной формы команды (no) выключает передачу соответствующей опции.

Синтаксис

```
next-server <ADDR>
no next-server
```

Параметры

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно задать только один адрес.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# next-server 172.16.32.1
```

option

Данная команда позволяет задать опции DHCP.

Использование отрицательной формы удаляет заданную опцию.

Синтаксис

```
option <CODE> { ip-address <ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
no option { <CODE> | all }
```

Параметры

<CODE> – код опции DHCP, задается в форме числа [0..255].

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 255 байт.

<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 15 IP-адресов, список задаётся через запятую.

<STRING> – текст, задаётся строкой от 1 до 255 символов.

all – удаляет все опции.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# option 43 ip-address 10.10.1.1
```

show ip dhcp binding

Данная команда позволяет посмотреть выданные DHCP-сервером IP-адреса. В примере ниже во второй колонке отображается значение опции 61 DHCP, указанной DHCP-клиентом, а в случае её отсутствия указывается MAC-адрес.

Синтаксис

```
show ip dhcp binding [ <ADDR> ] [vrf <VRF>]
```

Параметры

<ADDR> – IP-адрес, опциональный параметр, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Если в команде задан параметр <ADDR>, то будет отображена информация связанная только с указанным адресом.

<VRF> – имя экземпляра VRF, по которому будет выводиться информация. Задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip dhcp binding
 IP address         MAC / Client ID                                                  Binding type   Lease expires at
 ----------------   --------------------------------------------------------------   ------------   --------------------
 172.16.1.11        0x00656c7465782d636339642e613237312e373138342d6772655f31         active         2024-04-04 12:02:25
 172.16.1.12        0x00656c7465782d613866392e346261622e643730302d6772655f31         active         2024-04-04 12:02:27
 192.168.1.2        cc:9d:a2:71:71:c9                                                active         2024-04-04 17:04:27
 192.168.1.3        a8:f9:4b:ab:d7:01                                                active         2024-04-04 17:04:30
```

show ip dhcp server dscp

Данная команда позволяет посмотреть значение DSCP для сообщений DHCP-сервера.

Синтаксис

```
show ip dhcp server dscp
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt # show ip dhcp server dscp
DSCP:    32
```

show ip dhcp server pool

Данная команда позволяет посмотреть настроенные пулы IP-адресов. При указании имени выводится информация только для заданного пула.

Синтаксис

```
show ip dhcp server pool [ <POOL_NAME> ]
```

Параметры

<POOL_NAME> – имя пула, опциональный параметр.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip dhcp server pool lan-pool
name:                       lan-pool
network:                    192.168.1.0/24
address-ranges:             192.168.1.2-192.168.1.254
default-router:             192.168.1.1
max lease time (d:h:m):     001:0:0 (day:hour:min)
default lease time (d:h:m): 000:12:0 (day:hour:min)
```

show ip dhcp server vendor-specific

Данная команда позволяет посмотреть настроенные DHCP опции 43 и 60.

Синтаксис

```
show ip dhcp server vendor-specific
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip dhcp server vendor-specific
Vendor ID          Vendor options
---------------    --------------------------------
 Rusteletech             0x0b0931302e312e39302e32
```

suboption

Данная команда позволяет задать подопцию опции 43.

Использование отрицательной формы команды (no) удаляет заданную подопцию.

Синтаксис

```
suboption <CODE> {ascii-text <TEXT> | hex-bytes <HEX>}
no suboption <CODE> | all
```

Параметры

<CODE> – код подопции, задается в форме числа [0..255];

<STRING> – текст, задаётся строкой от 1 до 255 символов;

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 255 байт;

all – удаляет все опции.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER-VENDOR-SPECIFIC

Пример

```
rtt(config-dhcp-server-vendor-specific)# suboption 15 ascii-text
"http://192.168.1.1:8042"
```

tftp-server

Данная команда позволяет задать IP-адрес TFTP-сервера. Адрес передаётся клиентам в составе DHCP-опции 150.

Использование отрицательной формы команды (no) удаляет установленный IP-адрес TFTP-сервера.

Синтаксис

```
tftp-server <ADDR>
no tftp-server
```

Параметры

<ADDR> – IP-адрес TFTP-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

config-dhcp-server

Пример

```
rtt(config-ipv6-dhcp-server)# tftp-server 192.168.0.1
```

vendor-specific

Данная команда позволяет задать специфическую информацию поставщика (DHCP опция 43).

Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

Синтаксис

```
vendor-specific
no vendor-specific
```

Параметры

Отсутствуют.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-SERVER

Пример

```
rtt(config-dhcp-server)# vendor-specific
rtt(config-dhcp-server-vendor-specific)#
```

vendor-specific-options

Данная команда позволяет задать специфическую информацию поставщика (DHCP опция 43).

Использование отрицательной формы команды (no) удаляет специфическую информацию поставщика.

Синтаксис

```
vendor-specific-options <HEX>
no vendor-specific-options
```

Параметры

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-DHCP-VENDOR-ID

Пример

```
rtt(config-dhcp-vendor-id)# vendor-specific-options 0b0931302e312e39302e320
```

28.6. Настройка и мониторинг IPv6 DHCP-сервера

address

Данная команда позволяет добавить IPv6-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный IPv6-адрес.

Синтаксис

```
address <IPV6-ADDR> mac-address <MAC>
no address <IPV6-ADDR>
```

Параметры

<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если использовать команду для удаления, то при указании значения «all» будут удалены все IPv6-адреса;

<MAC> – MAC-адрес клиента, которому будет выдан IPv6-адрес, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# address fc00::2 mac-address A8:F9:4B:AA:00:40
```

address-range

Данная команда позволяет добавить диапазон IPv6-адресов к пулу адресов конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IPv6-адресов.

Синтаксис

```
[no] address-range <FROM-ADDR>-<TO-ADDR>
no address-range all
```

Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Можно указать до 32 диапазонов IPv6-адресов, список задаётся через запятую.

all – удалить все сконфигурированные диапазоны IPv6-адресов.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# address-range fc00::1-fc00:12,fc00::15-fc00::25
```

default-lease-time

Данная команда позволяет задать время аренды, на которое клиенту будет выдан IPv6-адрес, если клиент не запрашивал определенное время аренды.

Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
default-lease-time <TIME>
no default-lease-time
```

Параметры

<TIME> – время аренды IP-адреса, в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

Значение по умолчанию

12 часов.

Необходимый уровень привилегий

1

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# default-lease-time 00:04:00
```

dns-server

Данная команда позволяет задать список IPv6-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды удаляет указанный DNS-сервер из списка.

Синтаксис

```
dns-server <IPV6-ADDR>
no dns-server { <IPV6-ADDR> | all }
```

Параметры

<IPV6-ADDR> – IPv6-адрес DNS-сервера, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IPv6-адресов, список задаётся через запятую. Если использовать команду для удаления, то при указании значения «all» будут удалены все DNS-серверы.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# dns-server 8.8.8.8,8.8.4.4
```

domain-name

Данная команда позволяет задать имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опций.

Использование отрицательной формы команды (no) удаляет установленное имя домена.

Синтаксис

```
domain-name <NAME>
no domain-name
```

Параметры

<NAME> – имя домена клиента, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# domain-name Rusteletech.loc
```

excluded-address-range

Данная команда позволяет исключить диапазон IPv6-адресов из пула адресов, конфигурируемого DHCP-сервера.

Использование отрицательной формы команды (no) удаляет указанный диапазон IPv6-адресов.

Синтаксис

```
[no] excluded-address-range <FROM-ADDR>-<TO-ADDR>
```

Параметры

<FROM-ADDR> – начальный IPv6-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];

<TO-ADDR> – конечный IPv6-адрес диапазона, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

Можно указать до 32 диапазонов IPv6-адресов, список задаётся через запятую.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# excluded-address-range 10::1-10::9
```

ipv6 dhcp-server

Данная команда включает IPv6 DHCP-сервер.

Использование отрицательной формы команды (no) выключает IPv6 DHCP-сервер.

Синтаксис

```
[no] ipv6 dhcp-server [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать DHCP-сервер.

Значение по умолчанию

Выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 dhcp-server
```

ipv6 dhcp-server pool

Команда используется для создания пула IPv6-адресов DHCP-сервера и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный пул IPv6-адресов.

Синтаксис

```
[no] ipv6 dhcp-server pool <NAME> [vrf <VRF>]
```

Параметры

<NAME> – имя пула IPv6-адресов DHCP-сервера, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все пулы IPv6-адресов;

<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать данный пул IP-адресов DHCP-сервера.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 dhcp-server pool lan
```

ipv6 dhcp-server vendor-class-id

Данная команда необходима для создания идентификатора класса поставщика (DHCP опция 60) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный идентификатор класса поставщика.

Синтаксис

```
[no] ipv6 dhcp-server vendor-class-id <NAME>
```

Параметры

<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа. При выполнении отрицательной формы команды со значением параметра «all» будут удалены все идентификаторы класса поставщика.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ipv6 dhcp-server vendor-class-id Rusteletech
```

max-lease-time

Данная команда позволяет задать максимальное время аренды IPv6-адресов. Если IPv6 DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой. Использование отрицательной формы команды устанавливает значение по умолчанию.

Синтаксис

```
max-lease-time <TIME>
no max-lease-time
```

Параметры

<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

- DD – количество дней, принимает значения [0..364];
- HH – количество часов, принимает значения [0..23];
- MM – количество минут, принимает значения [0..59].

Значение по умолчанию

1 день.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# max-lease-time 00:16:00
```

network

Данная команда задает IPv6-адрес и маску для подсети, из которой будет выделен пул IP-адресов.

Использование отрицательной формы команды (no) удаляет настройки подсети в пуле.

Синтаксис

```
network <IPV6-ADDR/LEN>
no network
```

Параметры

<IPV6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X::X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# network fc00::/110
```

option

Данная команда позволяет задать опции DHCP.

Использование отрицательной формы удаляет данную опцию.

Синтаксис

```
option <CODE> { ip-address <IPV6-ADDR> | ascii-text <STRING> | hex-bytes <HEX> }
no option { <CODE> | all }
```

Параметры

<CODE> – код опции DHCP, задается в форме числа [1..255];

<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 255 символов;

<IPV6-ADDR> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IP-адресов, список задаётся через запятую;

<STRING> – текст, задаётся строкой от 1 до 255 символов;

all – удаляет все опции.

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IPV6-DHCP-SERVER

Пример

```
rtt(config-ipv6-dhcp-server)# option 43 ip-address fc00::/110
```

show ipv6 dhcp binding

Данная команда позволяет посмотреть выданные DHCP-сервером IPv6-адреса.

Синтаксис

```
show ipv6 dhcp binding [ <IPV6-ADDR> ]
```

Параметры

<IPV6-ADDR> – IPv6-адрес, опциональный параметр, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Если в команде задан параметр <IPV6-ADDR>, то будет отображена информация, связанная только с указанным адресом.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 dhcp binding
IPv6 address                                      Lease expires at
------------------------------------------------  --------------------
2001:db8:0:1::18                                  2015-06-09 23:39:45
```

show ipv6 dhcp server pool

Данная команда позволяет посмотреть настроенные пулы IPv6-адресов. При указании имени выводится информация только для заданного пула.

Синтаксис

```
show ipv6 dhcp server pool [ <POOL_NAME> ]
```

Параметры

<POOL_NAME> – имя пула, опциональный параметр, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 dhcp server pool lan_pool_ipv6
Name:                         lan_pool_ipv6
Network:                      2001:abcd:dcba::/120
Address-ranges:               2001:abcd:dcba::2-2001:abcd:dcba::10
Excluded-address-ranges:      --
Addresses:                    --
Dns-server:                   --
Options:                      --
Max lease time (d:h:m):       001:00:00
Default lease time (d:h:m):   000:12:00
```

show ipv6 dhcp server vendor-specific

Данная команда позволяет посмотреть настроенные DHCP опции 43 и 60.

Синтаксис

```
show ipv6 dhcp server vendor-specific
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ipv6 dhcp server vendor-specific
Vendor ID          Vendor options
----------------   ------------------------------
```

```
Rusteletech                0x0b0931302e312e39302e32
```

29.НАСТРОЙКА SLA

29.1. Общие команды настройки SLA

ip sla

Данной командой активируется функционал IP-SLA (SLA-agent).

Использование отрицательной формы команды (no) отключает функционал IP-SLA.

Синтаксис

```
[no] ip sla
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla
```

ip sla responder cisco

Данной командой на интерфейсе активируется функционал SLA-responder для Cisco-SLA-agent.

Использование отрицательной формы команды (no) на интерфейсе деактивируется функционал SLA-responder для Cisco-SLA-agent.

Синтаксис

```
[no] ip sla responder cisco
```

Параметры

Отсутствуют

Значение по умолчанию

Отключено

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# ip sla responder cisco
```

ip sla responder cisco port

Данной командой устанавливается UDP-порт, на котором будет идти прослушивание запросов от Cisco-SLA-agent.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ip sla responder cisco port <PORT>
no ip sla responder cisco port
```

Параметры

<PORT> – номер UDP-порта, задается в диапазоне [1..65535].

Значение по умолчанию

Не ограничено

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config-if-gi)# ip sla responder cisco port 7777
```

show ip sla configuration

Данной командой выводится информация о текущих настройках SLA-agent.

Синтаксис

```
show ip sla configuration
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip sla configuration
Mode:                          Rusteletech
State:                         Enabled
Logging:
  Delay:                       Enabled
  Error:                       Enabled
  Jitter:                      Enabled
  Losses:                      Enabled
  Status:                      Enabled
```

show ip sla test configuration

Данной командой выводится информация о сконфигурированных SLA-тестах.

Синтаксис

```
show ip sla test configuration [ <NUM> | vrf <VRF> ]
```

Параметры

<NUM> – номер SLA-теста, задается в диапазоне [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip sla test configuration
```

Test	State	Source	Destination	Freq	Interval	Packets	PacketSize	Timeout	Description
1	Enabled	192.168.44.11	192.168.44.13	10	20	100	74	3000	DNS reachability
2	Enabled	192.168.44.11	192.168.44.1	10	20	100	74	3000	Channel to Gateway
3	Enabled	192.168.44.11	192.168.44.13	10	20	100	74	3000	Channel to DNS-server

```
rtt# show ip sla test configuration 2

Test number:                        2
Description:                        Channel to Gateway
Test type:                          udp-jitter
State:                              Enabled
Authentication:                     Disabled
Destination address:                192.168.44.13
Destination port:                   1
Disallow fragmentation:             Disabled
Frequency:                          10
Interval:                           20
Number of packets:                  100
Packet size:                        74
Source address:                     192.168.44.1
Source interface:                   --
Source port:                        1
DSCP:                               0
Control phase destination port:     1800
Control phase source port:          --
Timeout:                            3000
Number of history records:          10
```

show ip sla test history

Данной командой выводится информация об истории измерений SLA-теста.

Синтаксис

```
show ip sla test history [ <NUM> | vrf <VRF> ] [ microseconds ] [ status ]
```

Параметры

<NUM> – номер SLA-теста, принимает значение [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа;

microseconds – ключ для отображения истории измерений в микросекундах;

status – ключ для отображения истории статусов SLA-теста.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip sla test history 3
```

Probe	TP	LP	LPF	LPR	OWDFA	OWDRA	OWJFA	OWJRA	TWDA	TWJA	DP	OSF	OSR	Last Run
1	100	0	0	0	0.17	0.17	0.01	0.01	0.35	0.03	0	0	0	2024-06-20 11:21:02
2	100	0	0	0	0.17	0.17	0.01	0.01	0.35	0.03	0	0	0	2024-06-20 11:20:52
3	100	0	0	0	0.17	0.17	0.01	0.01	0.35	0.03	0	0	0	2024-06-20 11:20:42
4	100	0	0	0	0.17	0.17	0.01	0.01	0.34	0.03	0	0	0	2024-06-20 11:20:32
5	100	0	0	0	0.17	0.17	0.01	0.01	0.35	0.03	0	0	0	2024-06-20 11:20:22
6	100	0	0	0	0.18	0.18	0.01	0.01	0.35	0.03	0	0	0	2024-06-20 11:20:11
7	100	0	0	0	0.17	0.17	0.01	0.01	0.34	0.03	0	0	0	2024-06-20 11:20:02
8	100	0	0	0	0.17	0.17	0.01	0.01	0.34	0.03	0	0	0	2024-06-20 11:19:52
9	100	0	0	0	0.17	0.17	0.01	0.01	0.34	0.03	0	0	0	2024-06-20 11:19:42
10	100	0	0	0	0.17	0.17	0.01	0.01	0.34	0.03	0	0	0	2024-06-20 11:19:32

show ip sla test statistics

Данной командой выводится информация об актуальных результатах измерений SLA-теста.

Синтаксис

```
show ip sla test statistics [ <NUM> | vrf <VRF> ] [ microseconds ]
```

Параметры

<NUM> – номер SLA-теста, принимает значение [1..10000];

<VRF> – имя VRF, задается строкой до 31 символа;

microseconds – ключ для отображения статистики в микросекундах.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip sla test statistics
```

Test	TP	LP	LPF	LPR	OWDFA	OWDRA	OWJFA	OWJRA	TWDA	TWJA	DP	OSF	OSR
1	100	--	--	--	--	--	--	--	0.35	0.05	--	--	--
2	100	0	0	0	0.19	0.19	0.04	0.04	0.38	0.08	0	0	0
3	100	0	0	0	0.18	0.18	0.02	0.02	0.35	0.03	0	0	0

```
rtt# show ip sla test statistics 2
Test number:                                    2
Description:                                    Channel to Gateway
Test status:                                    Successful
Transmitted packets:                            100
Lost packets:                                   0 (0.00%)
Lost packets in forward direction:              0 (0.00%)
Lost packets in reverse direction:              0 (0.00%)
One-way delay forward min/avg/max:              0.13/0.19/0.32 milliseconds
One-way delay reverse min/avg/max:              0.13/0.19/0.32 milliseconds
One-way jitter forward:                         0.04 milliseconds
One-way jitter reverse:                         0.04 milliseconds
Two-way delay min/avg/max:                      0.27/0.38/0.64 milliseconds
Two-way jitter min/avg/max:                     0.02/0.08/0.49 milliseconds
Duplicate packets:                              0
Out of sequence packets in forward direction: 0
Out of sequence packets in reverse direction: 0
Number of successes:                            10 (100.00%)
Number of failures:                             0 (0.00%)
```

show ip sla test status

Данной командой выводится сводная информация о статусах всех активных SLA-тестов.

Синтаксис

```
show ip sla test status
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show ip sla test status
```

```
Test      Type           Source           Destination      Status       Last Run            Description
-------   ------------   --------------   --------------   ----------   -----------------   -------------------
1         icmp-echo      192.168.44.11    192.168.44.13    Successful   2024-06-20 10:25:13   DNS reachability
2         udp-jitter     192.168.44.12    192.168.44.1     Successful   2024-06-20 10:25:14   Channel to Gateway
3         icmp-jitter    192.168.44.13    192.168.32.13    Successful   2024-06-20 10:25:11   Channel to DNS-server
```

29.2. Настройка Eltex-SLA

authentication algorithm

Данной командой указывается алгоритм для аутентификации приходящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет алгоритм для аутентификации приходящих запросов от SLA-agent.

Синтаксис

```
authentication algorithm <ALGORITHM>
no authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм аутентификации: sha-256, hmac-sha-256.

Значение по умолчанию

Не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-RESPONDER

Пример

```
rtt(config-sla-responder)# authentication algorithm sha-256
```

authentication key-chain

Данной командой указывается набор ключей для аутентификации приходящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет набор ключей для аутентификации приходящих запросов от SLA-agent.

Синтаксис

```
authentication key-chain <KEYCHAIN>
```

```
no authentication key-chain
```

Параметры

<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.

Значение по умолчанию

Не задан.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-RESPONDER

Пример

```
rtt(config-sla-responder)# authentication key-chain sla-chain
```

authentication key-string

Данной командой указывается пароль для аутентификации приходящих запросов от SLA-agent.

Использование отрицательной формы команды (no) удаляет пароль для аутентификации приходящих запросов от SLA-agent.

Синтаксис

```
authentication key-string ascii-text { <CLEAR-TEXT> | encrypted <ENCRYPTED-TEXT> }
no authentication key-string
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;
<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-RESPONDER

Пример

```
rtt(config-sla-responder)# authentication key-string ascii-text aukey
```

control-phase authentication algorithm

Данной командой задается алгоритм аутентификации для SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
control-phase authentication algorithm <ALGORITHM>
no control-phase authentication algorithm
```

Параметры

<ALGORITHM> – алгоритм хеширования, принимает значения [sha-256, hmac-sha-256].

Значение по умолчанию

Без шифрования

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase authentication algorithm hmac-sha-256
```

control-phase authentication key-id

Данной командой выбирается номер ключа в key-chain, сконфигурированном командой ip sla key-chain (см. раздел authentication key-chain), который будет отправляться для аутентификации на SLA-responder.

Использование отрицательной формы команды (no) удаляет номер ключа.

Синтаксис

```
control-phase authentication key-id <ID>
no control-phase authentication key-id
```

Параметры

<ID> – идентификатор ключа, задается в диапазоне [1..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase authentication key-id 2
```

control-phase authentication key-string

Данной командой задается ключ аутентификации, который будет отправляться для аутентификации на SLA-responder.

Использование отрицательной формы команды (no) удаляет ключ аутентификации, который будет отправляться для аутентификации на SLA-responder.

Синтаксис

```
control-phase authentication key-string ascii-text { <CLEAR-TEXT> |encrypted
<ENCRYPTED-TEXT>}
no control-phase authentication key-string
```

Параметры

<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;

<ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase authentication key-string ascii-text conphkey
```

control-phase destination-port

Данной командой устанавливается UDP-порт, на который будут отправляться запросы контрольной фазы на Eltex-SLA-responder.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
control-phase destination-port <PORT>
no control-phase destination-port
```

Параметры

<PORT> – номер UDP-порта, принимает значение [1..65535].

Значение по умолчанию

1800

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase destination-port 9999
```

control-phase retry

Данной командой устанавливается периодичность попыток установления контрольной фазы SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
control-phase retry <TIME>
no control-phase retry
```

Параметры

<TIME> – время ожидания перед повторной попыткой установления контрольной фазы SLA-теста, определяется в секундах [1..86400].

Значение по умолчанию

1

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase retry 300
```

control-phase source-port

Данной командой устанавливается UDP-порт, с которого будут отправляться запросы контрольной фазы на Eltex-SLA-responder.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
control-phase source-port <PORT>
no control-phase source-port
```

Параметры

<PORT> – номер UDP-порта, принимает значение [1..65535].

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase source-port 3333
```

control-phase timeout

Данной командой устанавливается время ожидания ответа от SLA-responder, по истечении которого контрольная фаза SLA-теста будет считаться неудачной.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
control-phase timeout <TIME>
no control-phase timeout
```

Параметры

<TIME> – время в миллисекундах, отведенное на прохождение контрольной фазы SLA-теста, принимает значение [1..4294967295].

Значение по умолчанию

3000

Необходимый уровень привилегий

15

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# control-phase timeout 5000
```

description

Данной командой задается описание SLA-теста.

Использование отрицательной формы команды (no) удаляет описание SLA-теста.

Синтаксис

```
description <DESCRIPTION>
no description
```

Параметры

<DESCRIPTION> – описание SLA-теста, задается строкой до 255 символов.

Значение по умолчанию

Отсутствует

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# description "Local gateway reachability"
```

disallow-fragmentation

Данная команда выставляет бит DF (don't fragment) в IP-заголовке равным единице для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] disallow-fragmentation
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

DF бит не установлен (равен 0)

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# disallow-fragmentation
```

dscp

Данной командой задается значение DSCP для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
dscp <DSCP>
no dscp
```

Параметры

<DSCP> – значение кода DSCP, принимает значение [0..63].

Значение по умолчанию

0

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# dscp 58
```

enable

Данной командой активируется SLA-тест.

Использование отрицательной формы команды (no) деактивирует SLA-тест.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# enable
```

frequency

Данной командой устанавливается период между перезапусками SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
frequency <TIME>
no frequency
```

Параметры

<TIME> – время в секундах, принимает значение [1..604800].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# frequency 3600
```

history

Данной командой задается количество сохраняемых результатов SLA-теста.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
history <SIZE>
no history
```

Параметры

<SIZE> – число сохраняемых результатов, принимает значение [1..1000].

Значение по умолчанию

10

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# history 100
```

icmp-echo

Данной командой задаётся icmp-режим тестирования канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

Синтаксис

```
icmp-echo { <DST-ADDRESS | IPV6-DST-ADDRESS> } { source-ip { <SRC-ADDRESS | IPV6-
SRC-ADDRESS> | object-group <NETWORK_OBJ_GROUP_NAME> } | source-interface { <IF> |
<TUN> } } [ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
no icmp-echo
```

Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-DST-ADDRESS> – IPv6-адрес SLA-responder. IP-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-SRC-ADDRESS> – IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..6000] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..100000].

Значение по умолчанию

<INTERVAL> – 20;

<NUM-PACKETS> – 100.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# icmp-echo 10.10.1.1 source-ip 192.168.54.22 num-packets 50
interval 15
```

icmp-jitter

Данной командой задаётся icmp-режим тестирования качества канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

Синтаксис

```
icmp-jitter <DST-ADDRESS> { source-ip <SRC-ADDRESS> | source-interface { <IF> |
<TUN> } } [ interval <INTERVAL> ] [ num-packets <NUM-PACKETS> ]
no icmp-echo
```

Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..6000] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..100000].

Значение по умолчанию

<INTERVAL> – 20;

<NUM-PACKETS> – 100.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# icmp-jitter 10.10.1.1 source-ip 192.168.54.22 num-packets 300
interval 100
```

ip sla key-chain

Данная команда определяет набор паролей для аутентификации через алгоритм хеширования md5 с SLA-responder/agent.

Использование отрицательной формы команды (no) удаляет привязку к набору паролей.

Синтаксис

```
ip sla key-chain <NAME>
no ip sla key-chain
```

Параметры

<NAME> – имя, задаётся строкой от 1 до 16 символов.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla key-chain KEY10
```

ip sla logging

Данной командой активируется логирование групп событий SLA.

Использование отрицательной формы команды (no) отключает логирование групп событий SLA.

Синтаксис

```
[no] ip sla logging <TYPE>
```

Параметры

<TYPE> – название группы информационных сообщений, может принимать значения:

- error – отображение сообщений об ошибках в работе SLA-тестов, причинах их провала, а также ошибок в работе SLA-responder (если таковой сконфигурирован в системе);
- delay – отображение сообщений о превышении/нормализации значений, установленных в thresholds delay;
- jitter – отображение сообщений о превышении/нормализации значений, установленных в thresholds jitter;
- losses – отображение сообщений о превышении/нормализации значений, установленных в thresholds losses;
- status – отображение сообщений о смене статуса SLA-теста.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla logging status
rtt(config)# ip sla logging error
```

ip sla responder

Данной командой выполняется переход в режим конфигурирования параметров SLA-responder.

Использование отрицательной формы команды (no) удаляет примененные ранее параметры SLA-responder.

Синтаксис

```
[no] ip sla responder [ vrf <VRF> ]
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа. При указании данного параметра SLA-responder включается в указанном VRF.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla responder
```

ip sla responder eltex

Данной командой на интерфейсе активируется функционал SLA-responder для Eltex-SLA-agent.

Использование отрицательной формы команды (no) на интерфейсе деактивируется функционал SLA-responder для Eltex-SLA-agent.

Синтаксис

```
[no] ip sla responder eltex
```

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config)# ip sla responder eltex
```

ip sla responder eltex port

Данной командой устанавливается UDP-порт, на котором будет идти прослушивание запросов прохождения контрольной фазы от Eltex-SLA-agent.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ip sla responder eltex port <PORT>
no ip sla responder eltex port
```

Параметры

<PORT> – номер UDP-порта, задается в диапазоне [1..65535].

Значение по умолчанию

1800

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config)# ip sla responder eltex port 5555
```

ip sla schedule

Данной командой устанавливается планировщик расписания работы SLA-тестов. Если SLA-тест создан и активирован, для него необходимо задать расписание.

Использование отрицательной формы команды (no) удаляет расписание работы SLA-тестов.

Синтаксис

```
ip sla schedule { <TEST-NUMBER> | all } [ life { <LIFE-TIME> | forever } ] [ start-time { <TIME> <MONTH> <DAY> | now } ]
[no] ip sla schedule { <TEST-NUMBER> | all }
```

Параметры

<TEST-NUMBER> – номер SLA-теста, принимает значение [1..10000]. При использовании ключа "all" вместо номера, расписание устанавливается для всех созданных SLA-тестов;

<LIFE-TIME> – время жизни теста, принимает принимает значение [1..2147483647] секунд;

forever – время жизни теста не ограничено;

<TIME> – время начала теста, задаётся в виде HH:MM:SS, где:

- HH – часы, принимает значение [0..23];
- MM – минуты, принимает значение [0..59];
- SS – секунды, принимает значение [0..59].

<MONTH> – месяц начала теста, принимает значения [January / February / March / April / May / June / July / August / September / October / November / December];

<DAY> – день месяца начала теста, принимает значения [1..31];

now – начать тест немедленно.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla schedule 1 life forever start-time now
```

ip sla test

Данная команда используется для создания SLA-теста (SLA-agent) и перехода в режим его конфигурирования.

Использование отрицательной формы команды (no) удаляет указанный SLA-тест.

Синтаксис

```
[no] ip sla test <NUM>
```

Параметры

<NUM> – номер SLA-теста, принимает значение [1..10000].

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# ip sla test 10
```

ipv6 sla responder eltex

Данной командой на интерфейсе активируется функционал SLA-responder для Eltex-SLA-agent.

Использование отрицательной формы команды (no) на интерфейсе деактивируется функционал SLA-responder для Eltex-SLA-agent.

Синтаксис

[no] ipv6 sla responder eltex

Параметры

Отсутствуют.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config)# ip sla responder eltex
```

ipv6 sla responder eltex port

Данной командой устанавливается UDP-порт, на котором будет идти прослушивание запросов прохождения контрольной фазы от Eltex-SLA-agent.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

ipv6 sla responder eltex port <PORT>

no ipv6 sla responder eltex port

Параметры

<PORT> – номер UDP-порта, задается в диапазоне [1..65535].

Значение по умолчанию

1800

Необходимый уровень привилегий

10

Командный режим

CONFIG-IF-GI

CONFIG-IF-TE

CONFIG-IF-TWE

CONFIG-IF-FO

CONFIG-IF-HU

CONFIG-IF-SUB

CONFIG-IF-QINQ

CONFIG-IF-PORT-CHANNEL

CONFIG-BRIDGE

CONFIG-GRE

CONFIG-IP4IP

CONFIG-L2TP

CONFIG-PPTP

CONFIG-VTI

CONFIG-IF-E1

CONFIG-IF-MULTILINK

CONFIG-LOOPBACK

Пример

```
rtt(config)# ip sla responder eltex port 5555
```

packet-size

Данной командой задается размер исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
packet-size <SIZE>
no packet-size
```

Параметры

<SIZE> – размер тестовых пакетов SLA-теста, принимает значение [70..18000] байт.

Значение по умолчанию

74

90 (для ipv6 udp)

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# packet-size 256
```

tcp-connect

Данной командой задаётся tcp-режим тестирования качества канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

Синтаксис

```
tcp-connect { <DST-ADDRESS> | <IPV6-DST-ADDRESS> } <DST-PORT> { source-ip { <SRC-
ADDRESS> | <IPV6-SRC-ADDRESS> | object-group <NETWORK_OBJ_GROUP_NAME> } | source-
interface { <IF> | <TUN> } } [ source-port <SRC-PORT> ]
no tcp-connect
```

Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-DST-ADDRESS> – IPv6-адрес SLA-responder. IP-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<DST-PORT> – номер TCP-порта назначения тестовых пакетов, принимает значения [1..65535];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-SRC-ADDRESS> - IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<SRC-PORT> – номер UDP-порта источника тестовых пакетов, принимает значения [1..65535].

Значение по умолчанию

<SRC-PORT> – любой свободный.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# tcp-connect 192.168.0.1 23 source-interface gigabitethernet
1/0/1 source-port 64550
```

thresholds delay

Данной командой устанавливаются допустимые значения задержек, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров задержки.

Синтаксис

```
thresholds delay { high <DELAY> | low <DELAY> | forward { high <DELAY> | low
<DELAY> } | reverse {high <DELAY> | low <DELAY> } } [ from-last <NUM-CHECK> { all |
over <NUM-LIMIT> } ]
no thresholds delay { high | low | forward { high | low } | reverse { high | low }
}
```

Параметры

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA- responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры задержки;

<DELAY> – значение задержки в миллисекундах, принимает значение в диапазоне [1..60000];

<NUM-CHECKS> – количество итераций теста, в течение которых проверяется соблюдение пороговых значений характеристик канала;

<NUM-LIMIT> – количество итераций теста с превышением установленного порога, после превышения которого тест считается проваленным.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# thresholds delay low 20
```

thresholds jitter

Данной командой устанавливаются допустимые значения джиттера, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров задержки.

Синтаксис

```
thresholds jitter { high <JITTER> | low <JITTER> | forward { high <JITTER> | low
<JITTER> } | reverse { high <JITTER> | low <JITTER> } } [ from-last <NUM-CHECK> {
all | over <NUM-LIMIT> } ]
no thresholds jitter { high | low | forward { high | low } | reverse { high | low }
}
```

Параметры

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA-responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры джиттера;

<JITTER> – значение джиттера в миллисекундах, принимает значение в диапазоне [1..60000];

<NUM-CHECKS> – количество итераций теста, в течение которых проверяется соблюдение пороговых значений характеристик канала;

<NUM-LIMIT> – количество итераций теста с превышением установленного порога, после превышения которого тест считается проваленным.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# thresholds jitter forward high 10
```

thresholds losses

Данной командой устанавливаются допустимые значения потерь, при пересечении которых тест считается проваленным.

Использование отрицательной формы команды (no) удаляет пороги и отключает отслеживание параметров потерь.

Синтаксис

```
thresholds losses { high <NUM-PACKETS> | low <NUM-PACKETS> | forward { high <NUM-
PACKETS> | low <NUM-PACKETS> } | reverse { high <NUM-PACKETS> | low <NUM-PACKETS> }
} [ from-last <NUM-CHECK> { all | over <NUM-LIMIT> } ]
no thresholds losses { high | low | forward | low } | reverse { high | low } }
```

Параметры

high – максимально допустимое значение параметра;

low – минимально допустимое значение параметра;

forward – измерение параметра в направлении от SLA-agent до SLA-responder;

reverse – измерение параметра в направлении от SLA-responder до SLA-agent;

Без указания forward/reverse – задаются суммарные параметры потерь;

<NUM-PACKETS> – количество пакетов, принимает значение в диапазоне [1..60000];

<NUM-CHECKS> – количество итераций теста, в течение которых проверяется соблюдение пороговых значений характеристик канала;

<NUM-LIMIT> – количество итераций теста с превышением установленного порога, после превышения которого тест считается проваленным.

Значение по умолчанию

Отключено.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# thresholds losses reverse high 10
```

timeout

Данной командой устанавливается время ожидания очередного пакета от SLA-responder, по истечении которого SLA-agent закроет SLA-сессию.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
timeout <TIME>
no timeout
```

Параметры

<TIME> – время ожидания, определяется в миллисекундах [1..4294967295].

Значение по умолчанию

3000

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# timeout 1000
```

timeout

Данной командой устанавливается время ожидания очередного пакета от SLA-agent, по истечении которого SLA-responder закроет SLA-сессию.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
timeout <TIME>
no timeout
```

Параметры

<TIME> – время. отведенное на прохождение SLA-теста, определяется в миллисекундах [1..4294967295].

Значение по умолчанию

3000

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-RESPONDER

Пример

```
rtt(config-sla-responder)# timeout
```

ttl

Данной командой задается значение TTL для исходящих на SLA-responder пакетов.

Использование отрицательной формы команды (no) устанавливается значение по умолчанию.

Синтаксис

```
ttl <TTL>
no ttl
```

Параметры

<TTL> – значение TTL, принимает значения в диапазоне [1..255].

Значение по умолчанию

64

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# ttl 10
```

udp-jitter

Данной командой задаётся UDP-режим тестирования канала связи и параметры тестирования.

Использование отрицательной формы команды (no) удаляет конфигурацию режима и параметров тестирования.

Синтаксис

```
udp-jitter { <DST-ADDRESS> | <IPV6-DST-ADDRESS> } <DST-PORT> { source-ip { <SRC-
ADDRESS> | <IPV6-SRC-ADDRESS> | object-group <NETWORK_OBJ_GROUP_NAME> } | source-
interface { <IF> | <TUN> } } [ source-port <SRC-PORT> ] [ interval <INTERVAL> ] [
num-packets <NUM-PACKETS> ]
no udp-jitter
```

Параметры

<DST-ADDRESS> – IPv4-адрес SLA-responder. IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-DST-ADDRESS> – IPv6-адрес SLA-responder. IP-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<DST-PORT> – номер UDP-порта назначения тестовых пакетов, принимает значения [1..65535];

<SRC-ADDRESS> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

<IPV6-SRC-ADDRESS> - IPv6-адрес, задаётся в виде X:X:X:X::X, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF];

<NETWORK_OBJ_GROUP_NAME> – список адресов, которые будут использоваться в качестве source address;

<IF> – тип и идентификатор интерфейса/туннеля, IP-адрес которого будет использоваться в качестве адреса источника пакетов, задаётся в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора;

<TUN> – имя туннеля устройства, задаётся в виде, описанном в разделе Типы и порядок именования туннелей маршрутизатора;

<SRC-PORT> – номер UDP-порта источника тестовых пакетов, принимает значения [1..65535];

<INTERVAL> – интервал отправки тестовых пакетов, принимает значение [1..255] миллисекунд;

<NUM-PACKETS> – количество тестовых пакетов, отправляемых в рамках теста, принимает значение [1..6000].

Значение по умолчанию

<INTERVAL> – 20

<NUM-PACKETS> – 100

<SRC-PORT> – любой свободный

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# udp-jitter 10.10.1.1 50000 source-ip 192.168.54.22 num-
packets 50 interval 15
```

vrf

Данной командой задается экземпляр VRF, в адресном пространстве которого должен работать SLA-тест.

Использование отрицательной формы команды (no) возвращает SLA-тест в адресное пространство глобального маршрутизатора.

Синтаксис

```
vrf <VRF>
no vrf
```

Параметры

<VRF> – имя экземпляра VRF, задаётся строкой до 31 символа.

Необходимый уровень привилегий

10

Командный режим

CONFIG-SLA-TEST

Пример

```
rtt(config-sla-test)# vrf subrouter
```

30. НАСТРОЙКА WI-FI КОНТРОЛЛЕРА ТУННЕЛЕЙ

Данный функционал поддерживается при наличии лицензии.

aaa das-profile

Данная команда используется для выбора профиля серверов динамической авторизации (DAS), на которые будут приходить CoA-запросы от PCRF об изменении политики обслуживания.

Использование отрицательной формы команды (no) удаляет заданный профиль серверов динамической авторизации (DAS).

Синтаксис

```
[no] aaa das-profile <NAME>
```

Параметры

<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# aaa das-profile profile1
```

aaa radius-profile

Данная команда используется для выбора профиля RADIUS-серверов, на которые будут отправляться запросы для получения конфигурации SoftGRE DATA-туннелей (обслуживаемых SSID и параметров шейпинга).

Использование отрицательной формы команды (no) удаляет заданный профиль RADIUS-серверов.

Синтаксис

```
[no] aaa radius-profile <NAME>
```

Параметры

<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# aaa radius-profile profile1
```

data-tunnel configuration

Данной командой устанавливается режим конфигурации SoftGRE DATA туннелей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
data-tunnel configuration { local | radius }
no data-tunnel configuration
```

Параметры

local – режим конфигурации, при котором параметры SoftGRE DATA туннелей получаются из локальной конфигурации маршрутизатора.

radius – режим, при котором параметры SoftGRE DATA туннелей запрашиваются у RADIUS-сервера.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# data-tunnel configuration radius
```

enable

Данной командой активируется контроллер Wi-Fi.

Использование отрицательной формы команды (no) деактивирует контроллер Wi-Fi.

Синтаксис

```
[no] enable
```

Параметры

Команда не содержит параметров.

Значение по умолчанию

Процесс выключен.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# enable
```

failure-count

Данной командой определяется количество последовательных неудачных ICMP-запросов после которых, при отсутствии ответа от встречной стороны, туннель считается нерабочим и удаляется из системы.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] failure-count <VALUE>
```

Параметры

<VALUE> – количество неудачных ping-запросов, принимает значения [1..100].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# failure-count 8
```

internal-usage-vlan

Данной командой определяется список VLAN, которые будут использоваться для формирования switch-community.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
[no] internal-usage-vlan <VLAN_List>
```

Параметры

<VLAN_List> – идентификационный номер VLAN, диапазон допустимых значений [2 .. 4094].

Можно указать несколько VLAN перечислением через запятую «,» либо указать диапазон VLAN через дефис «-».

Значение по умолчанию

Отсутствует.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# internal-usage-vlan 1001-2037,2500-3000,3200-3700
```

keepalive mode

Данной командой устанавливается режим работы механизма keepalive для SoftGRE-туннелей.

Использование отрицательной формы команды (no) устанавливает значение по умолчанию.

Синтаксис

```
keepalive mode { proactive | reactive }
no keepalive mode
```

Параметры

- proactive – режим проверки SoftGRE-туннелей через icmp-проверку удаленного адреса;
- reactive – режим проверки SoftGRE-туннелей через прием icmp-пакетов с payload: “ELTEX_GRE IPv4 < туннельный ip address устройства>” от удаленных устройств внутри туннеля.

Значение по умолчанию

proactive

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# keepalive mode reactive
```

keepalive-disable

Данная команда отключает обмен ICMP-сообщениями, которые используются для проверки доступности удаленного шлюза туннелей Wi-Fi контроллера.

Использование отрицательной формы команды (no) включает обмен ICMP-сообщениями.

Синтаксис

```
[no] keepalive-disable
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# keepalive-disable
```

nas-ip-address

Данной командой определяется IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых RADIUS-пакетах.

Использование отрицательной формы команды (no) удаляет указанный IP-адрес источника.

Синтаксис

```
nas-ip-address <ADDR>
no nas-ip-address
```

Параметры

<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

15

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# nas-ip-address 10.100.100.2
```

peer-address

Данной командой определяется IP-адрес соседа, с которым будет осуществляться резервирование туннелей.

Использование отрицательной формы команды (no) удаляет IP-адрес соседнего маршрутизатора из конфигурации.

Синтаксис

```
[no] peer-address <ADDR>
```

Параметры

<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# peer-address 192.168.0.15
```

resp-time

Данной командой определяется время ожидания ответа, после истечении которого ICMP-запрос считается отклоненным.

Использование отрицательной формы устанавливает значение resp-time по умолчанию.

Синтаксис

```
[no] resp-time <TIME >
```

Параметры

<TIME > – количество секунд, принимает значения [1..30].

Значение по умолчанию

5

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# resp-time 30
```

retry-time

Данной командой устанавливается время между ICMP-запросами.

Использование отрицательной формы команды (no) устанавливает значение retry-time по умолчанию.

Синтаксис

```
[no] retry time < TIME >
```

Параметры

< TIME > – количество секунд, принимает значения [60..3600].

Необходимый уровень привилегий

10

Значение по умолчанию

60

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# retry-time 12
```

show interfaces bridge switch-communities

Командой выполняется просмотр списка сформированных switch-community.

Синтаксис

show interfaces bridge switch-communities [<BRIDGE-ID>]

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces bridge switch-communities
bridge 10
Location                        Community                                  Interfaces
-------------   ---------------------------------------   ---------------
data10         Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.10, softgre 4.10, softgre
6.10,
               .MRF_Sibir.Local.AllClients.root          softgre 8.10

data10         TEST.Novosibirsk.Novosibirskaya_oblast    softgre 2.310, softgre 4.310,
softgre
               .MRF_Sibir.Local.AllClients.root          6.310, softgre 8.310


bridge 11
```

```
Location                          Community                                  Interfaces
-------------   ---------------------------------------   ---------------------
data11          Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.311, softgre 4.311,
softgre
                .MRF_Sibir.Local.AllClients.root          6.311, softgre 8.311
```

show interfaces bridge switch-communities community

Командой выполняется просмотр списка сформированных switch-community, соответствующих определенному названию домена СУ EMS.

Синтаксис

show interfaces bridge switch-communities community <EMS-DOMAIN> [<BRIDGE-ID>]

Параметры

<EMS-DOMAIN> – имя домена СУ EMS, WORD(1-246).

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces bridge switch-communities community
Eltex.Novosibirsk.Novosibirskaya_oblast.MRF_Sibir.Local.AllClients.root
bridge 10
Location       Community                                 Interfaces
------------   ---------------------------------------   --------------------------
data10         Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.10, softgre 4.10, softgre
6.10,
               .MRF_Sibir.Local.AllClients.root          softgre 8.10


bridge 11
Location                          Community                                  Interfaces
------------   ---------------------------------------   -----------------------------------
----
data11         Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.311, softgre 4.311,
softgre
               .MRF_Sibir.Local.AllClients.root          6.311, softgre 8.311
```

show interfaces bridge switch-communities community include

Командой выполняется просмотр списка сформированных switch-community, соответствующих определенному шаблону в названии домена СУ EMS.

Синтаксис

show interfaces bridge switch-communities community include <WORD> [<BRIDGE-ID>]

Параметры

<WORD> – шаблон, по которому выполняется поиск в имени домена СУ EMS, WORD(1-246).

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces bridge switch-communities community include Rusteletech
bridge 10
Location      Community                                 Interfaces
-----------   ---------------------------------------   ----------------------------------
------
data10        Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.10, softgre 4.10, softgre
6.10,
              .MRF_Sibir.Local.AllClients.root          softgre 8.10


bridge 11
Location      Community                                 Interfaces
-----------   ---------------------------------------   ----------------------------------
------
data11        Eltex.Novosibirsk.Novosibirskaya_oblast   softgre 2.311, softgre 4.311,
softgre
              .MRF_Sibir.Local.AllClients.root          6.311, softgre 8.311
```

show interfaces bridge switch-communities summary

Командой выполняется просмотр количества сформированных switch-community.

Синтаксис

show interfaces bridge switch-communities summary [<BRIDGE-ID>]

Параметры

<BRIDGE-ID> – идентификационный номер моста, задается в виде, описанном в разделе Типы и порядок именования интерфейсов маршрутизатора.

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show interfaces bridge switch-communities summary
Bridges      Communities
----------   -----------
bridge 10    2
bridge 11    1
```

show wireless-controller access-points

Команда вводит суммирующую информацию о подключенных ТД:

- MAC-адрес;
- туннельный адрес;
- адрес управления;
- сабтуннели.

Синтаксис

show wireless-controller access-points [mac <ADDR> | management-ip <ADDR> | remote-ip <ADDR>]

Параметры

mac <ADDR> – MAC-адрес ТД, по которому ведется поиск, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];

management-ip <ADDR> – IP-адрес управления, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];

remote-ip <ADDR> – IP-адрес, поднимающий туннель, по которому ведется поиск, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

Необходимый уровень привилегий

1

Командный режим

ROOT

Пример

```
rtt# show wireless-controller access-points
```

```
MAC adress          Remote IP         Management IP     Sub tunnels
------------------  ----------------  ----------------  ---------------------------------------
e0:d9:e3:73:07:61   100.124.0.36      10.102.14.216     softgre 1.1, softgre 2.12, softgre 2.14
a8:f9:4b:b0:2c:c1   100.124.0.34      10.102.29.229     softgre 3.1, softgre 4.12, softgre
                                                        4.122, softgre 4.321, softgre 4.322,
                                                        softgre 4.323, softgre 4.489, softgre
                                                        4.2408

a8:f9:4b:ac:ae:aa   100.124.0.231     10.102.16.73      softgre 5.1, softgre 6.12, softgre
                                                        6.122, softgre 6.321, softgre 6.322,
                                                        softgre 6.323, softgre 6.489, softgre
                                                        6.2408

00:00:2a:d9:d8:a7   10.10.45.2        10.102.17.248     softgre 7.1, softgre 8.12, softgre
                                                        8.2011, softgre 8.2013

00:00:2a:d9:d8:a5   10.10.45.1        10.102.18.115     softgre 9.1, softgre 10.12, softgre
                                                        10.2011, softgre 10.2013

00:00:2a:d9:d8:a9   10.10.45.3        10.102.18.196     softgre 11.1, softgre 12.12, softgre
                                                        12.2011, softgre 12.2013
```

```
rtt# show wireless-controller access-points remote-ip 100.124.0.231
MAC adress          Remote IP         Management IP     Sub tunnels
------------------  ----------------  ----------------  ---------------------------------------
a8:f9:4b:ac:ae:aa   100.124.0.231     10.102.16.73      softgre 5.1, softgre 6.12, softgre
                                                        6.122, softgre 6.321, softgre 6.322,
                                                        softgre 6.323, softgre 6.489, softgre
                                                        6.2408
```

thresholds tunnels-number

Данная команда используется для задания порога количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh и eltexInterfacesNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

Синтаксис

```
thresholds tunnels-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds tunnels-number { high | low }
```

Параметры

<TH-HIGH> – порог количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – порог количества туннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHighOk.

Значение по умолчанию

На R800 <TH-HIGH> – 900, <TH-LOW> – 850;

На R100/200 <TH-HIGH> – 400, <TH-LOW> – 350.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# thresholds tunnels-number high 1000
```

thresholds sub-tunnels-number

Данная команда используется для задания порога количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh и eltexInterfacesNumberHighOk.

Использование отрицательной формы команды (no) устанавливает значения по умолчанию.

Синтаксис

```
thresholds sub-tunnels-number { high <TH-HIGH> | low <TH-LOW> }
no thresholds sub-tunnels-number { high | low }
```

Параметры

<TH-HIGH> – порог количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHigh;

<TH-LOW> – порог количества сабтуннелей SoftGRE для отправки snmp-trap eltexInterfacesNumberHighOk.

Значение по умолчанию

На R800 <TH-HIGH> – 2500, <TH-LOW> – 2000;

На R100/200 <TH-HIGH> – 1000, <TH-LOW> – 900.

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# thresholds sub-tunnels-number high 2000
```

vrrp-group

Данной командой определяется VRRP-группа, на основе которой определяется состояние (основной/резервный) Wi-Fi контроллера.

Использование отрицательной формы команды (no) удаляет идентификатор VRRP.

Синтаксис

```
vrrp-group <GRID>
no vrrp-group
```

Параметры

<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].

Необходимый уровень привилегий

10

Командный режим

CONFIG-WIRELESS

Пример

```
rtt(config-wireless)# vrrp-group 10
```

wireless-controller

Переход в режим конфигурирования контроллера Wi-Fi.

Использование отрицательной формы команды (no) очищает конфигурацию и выключает контроллер Wi-Fi туннелей.

Синтаксис

```
[no] wireless-controller
```

Параметры

Команда не содержит параметров.

Необходимый уровень привилегий

10

Командный режим

CONFIG

Пример

```
rtt(config)# wireless-controller
```