



Маршрутизаторы серии RTT-R RTT-R100, RTT-R200, RTT-R800

Руководство по эксплуатации

версия ПО 1.4.4

Версия документа	Дата выпуска	Содержание изменений
Версия 1.14	28.06.2019	Изменения в разделах: - 7.31 Настройка клиента удаленного доступа по протоколу PPPoE - 7.32 Настройка клиента удаленного доступа по протоколу PPTP - 7.33 Настройка клиента удаленного доступа по протоколу L2TP
Версия 1.13	11.11.2018	Синхронизация с версией ПО 1.4.1
Версия 1.12	17.04.2018	Синхронизация с версией ПО 1.4.0
Версия 1.11	20.12.2017	Синхронизация с версией ПО 1.3.0
Версия 1.10	04.09.2017	Синхронизация с версией ПО 1.2.1
Версия 1.9	03.05.2017	Добавлены разделы: - 7.19.2 Настройка Policy-based IPsec VPN - 7.35 Настройка BRAS (Broadband Remote Access Server) Изменения в разделах: - 2.3 Основные технические характеристики - 2.4 Конструктивное исполнение - 2.5 Комплект поставки - 3.3 Установка модулей питания RTT-R800 - 5.1 Заводская конфигурация маршрутизатора RTT
Версия 1.8	14.12.2016	Добавлены разделы: - 7.2 Настройка терминирования Q-in-Q - 7.20 Настройка LT-туннелей - 7.31 Настройка VRRP tracking
Версия 1.7	03.03.2016	Добавлен раздел: - 8 Часто задаваемые вопросы
Версия 1.6	24.02.2016	Добавлены разделы: - 7.15.1 Настройка Route-map для BGP - 7.21 Настройка удаленного доступа к корпоративной сети по OpenVPN протоколу - 7.31 Настройка SNMP Изменения в разделах: - 7.15 Настройка политики маршрутизации PBR - 7.19 Настройка удаленного доступа к корпоративной сети по PPTP-протоколу
Версия 1.5	06.08.2015	Добавлено описание устройств RTT-100, RTT-200 Добавлен раздел: - 2.4.2 Конструктивное исполнение RTT-100, RTT-200 Изменения в разделах: - 2.4 Конструктивное исполнение - 2.5 Комплект поставки - 3 Установка и подключение - 7.1 Настройка VLAN - 7.6 Конфигурирование Source NAT - 7.16 Настройка L2TPv3-туннелей - 7.24 Настройка Netflow - 7.25 Настройка sFlow - 7.26 Настройка LACP
Версия 1.4	09.06.2015	Добавлены разделы: - 6.1 Настройка AAA - 6.1 Настройка привилегий пользователя - 6.7 Настройка списков доступа (ACL) - 6.9 Настройка MLPPP - 6.14 Настройка Route-map - 6.21.2 Расширенный QoS - 6.24 Настройка VRF Lite Изменения в разделах: - 2.4.4 Световая индикация

Версия 1.3	05.03.2015	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> - 6.15 Настройка Dual-Homing - 6.16 Настройка QoS - 6.17 Настройка зеркалирования - 6.18 Настройка VRRP - 6.19 Настройка MultiWAN <p>Изменения в разделах:</p> <ul style="list-style-type: none"> - 6.4 Конфигурирование firewall - 6.5 Конфигурирование статических маршрутов - 6.6 Настройка Bridge - 6.7 Настройка RIP - 6.8 Настройка OSPF - 6.9 Настройка BGP - 6.10 Настройка GRE-туннелей - 6.11 Настройка L2TPv3-туннелей - 6.12 Настройка Route-based IPsec VPN - 6.13 Настройка удаленного доступа к корпоративной сети по PPTP-протоколу - 6.14 Настройка удаленного доступа к корпоративной сети по L2TP/IPsec протоколу - 7.1 Обновление программного обеспечения средствами системы - 7.2 Обновление программного обеспечения из начального загрузчика
Версия 1.2	02.12.2014	<p>Добавлены разделы:</p> <ul style="list-style-type: none"> - 6.6 Настройка Bridge - 6.7 Настройка RIP - 6.8 Настройка OSPF - 6.9 Настройка BGP - 6.10 Настройка L3-туннелей (GRE) - 6.11 Настройка L2TPv3-туннелей (L2TPv3)
Версия 1.1	03.06.2014	<p>Добавлен раздел:</p> <ul style="list-style-type: none"> - 6 Настройка маршрутизатора.
Версия 1.0	25.04.2014	Первая публикация.
Версия программного обеспечения	1.4.4	

СОДЕРЖАНИЕ

1	ВВЕДЕНИЕ.....	8
1.1	Аннотация	8
1.2	Целевая аудитория.....	8
1.3	Условные обозначения	8
2	ОПИСАНИЕ ИЗДЕЛИЯ	10
2.1	Назначение	10
2.2	Функции.....	10
2.2.1	Функции интерфейсов.....	10
2.2.2	Функции при работе с MAC-адресами	11
2.2.3	Функции второго уровня сетевой модели OSI	11
2.2.4	Функции третьего уровня сетевой модели OSI	12
2.2.5	Функции туннелирования трафика	13
2.2.6	Функции управления и конфигурирования.....	13
2.2.7	Функции сетевой защиты.....	14
2.3	Основные технические характеристики	14
	Общие параметры	14
2.4	Конструктивное исполнение	16
2.4.1	Конструктивное исполнение RTT-R800.....	16
2.4.2	Конструктивное исполнение RTT-R100, RTT-R200.....	18
2.4.3	Световая индикация	20
2.5	Комплект поставки	23
3	УСТАНОВКА И ПОДКЛЮЧЕНИЕ	25
3.1	Крепление кронштейнов	25
3.2	Установка устройства в стойку	26
3.3	Установка модулей питания RTT-R800	27
3.4	Подключение питающей сети	27
3.5	Установка и удаление SFP-трансиверов	28
3.5.1	Установка трансивера.....	28
3.5.2	Удаление трансивера	28
4	ИНТЕРФЕЙСЫ УПРАВЛЕНИЯ	30
4.1	Интерфейс командной строки (CLI)	30
4.2	Типы и порядок именования интерфейсов маршрутизатора	30
4.3	Типы и порядок именования туннелей маршрутизатора.....	32
5	НАЧАЛЬНАЯ НАСТРОЙКА МАРШРУТИЗАТОРА	34
5.1	Заводская конфигурация маршрутизатора RTT	34
5.1.1	Описание заводской конфигурации	34
5.2	Подключение и конфигурирование маршрутизатора	35
5.2.1	Подключение к маршрутизатору	35
5.2.2	Применение изменения конфигурации	36
5.2.3	Базовая настройка маршрутизатора.....	36
6	ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ.....	40
6.1	Обновление программного обеспечения средствами системы.....	40
6.2	Обновление программного обеспечения из начального загрузчика.....	42
6.3	Обновление вторичного загрузчика (U-Boot)	43
7	ПРИМЕРЫ НАСТРОЙКИ МАРШРУТИЗАТОРА	44
7.1	Настройка VLAN	44
7.1.1	Алгоритм настройки	44
7.1.2	Пример настройки 1. Удаление VLAN с интерфейса	44
7.1.3	Пример настройки 2. Разрешение обработки VLAN в тегированном режиме	45
7.1.4	Пример настройки 3. Разрешение обработки VLAN в тегированном и не тегированном режиме	45
7.2	Настройка LLDP	46

7.2.1	Алгоритм настройки	46
7.2.2	Пример настройки	47
7.3	Настройка LLDP MED	48
7.3.1	Алгоритм настройки	48
7.3.2	Пример настройки Voice VLAN	49
7.4	Настройка терминации на саб-интерфейсе	50
7.4.1	Алгоритм настройки	51
7.4.2	Пример настройки саб-интерфейса	51
7.5	Настройка терминации на Q-in-Q-интерфейсе	52
7.5.1	Алгоритм настройки	52
7.5.2	Пример настройки q-in-q-интерфейса	53
7.6	Настройка USB модемов	53
7.6.1	Алгоритм настройки USB-модемов	54
7.6.2	Пример настройки	55
7.7	Настройка AAA	56
7.7.1	Алгоритм настройки локальной аутентификации	56
7.7.2	Алгоритм настройки AAA по протоколу RADIUS	59
7.7.3	Алгоритм настройки AAA по протоколу TACACS	62
7.7.4	Алгоритм настройки AAA по протоколу LDAP	65
7.7.5	Пример настройки аутентификации по telnet через RADIUS-сервер	68
7.8	Настройка привилегий команд	69
7.8.1	Алгоритм настройки	69
7.8.2	Пример настройки привилегий команд	70
7.9	Настройка DHCP-сервера	70
7.9.1	Алгоритм настройки	70
7.9.2	Пример настройки DHCP-сервера	73
7.10	Конфигурирование Destination NAT	75
7.10.1	Алгоритм настройки	75
7.10.2	Пример настройки Destination NAT	77
7.11	Конфигурирование Source NAT	79
7.11.1	Алгоритм настройки	79
7.11.2	Пример настройки 1	81
7.11.3	Пример настройки 2	83
7.12	Конфигурирование Static NAT	85
7.10.1	Алгоритм настройки	85
7.10.2	Пример настройки Static NAT	85
7.10.3	Пример настройки фильтрации приложений (DPI)	87
7.13	Настройка логирования и защиты от сетевых атак	88
7.13.1	Алгоритм настройки	88
7.13.2	Описание механизмов защиты от атак	91
7.13.3	Пример настройки логирования и защиты от сетевых атак	94
7.14	Конфигурирование Firewall	95
7.14.1	Алгоритм настройки	96
7.14.2	Пример настройки Firewall	101
7.15	Настройка списков доступа (ACL)	104
7.15.1	Алгоритм настройки	104
7.15.2	Пример настройки списка доступа	106
7.16	Конфигурирование статических маршрутов	106
7.16.1	Процесс настройки	107
7.16.2	Пример настройки статических маршрутов	108
7.17	Настройка MLPPP	109
7.17.1	Алгоритм настройки	110
7.17.2	Пример настройки	112

7.18	Настройка Bridge.....	113
7.18.1	Алгоритм настройки.....	113
7.18.2	Пример настройки bridge для VLAN и L2TPv3-туннеля	114
7.18.3	Пример настройки bridge для VLAN.....	115
7.18.4	Пример настройки добавления/удаления второго VLAN-тэга	116
7.19	Настройка RIP	117
7.19.1	Алгоритм настройки.....	117
7.19.2	Пример настройки RIP	120
7.20	Настройка OSPF.....	122
7.20.1	Алгоритм настройки.....	122
7.20.2	Пример настройки OSPF	129
7.20.3	Пример настройки OSPF stub area	130
7.20.4	Пример настройки Virtual link	131
7.21	Настройка BGP	132
7.21.1	Алгоритм настройки.....	133
7.21.2	Пример настройки.....	139
7.22	Настройка BFD.....	140
7.22.1	Алгоритм настройки.....	140
7.22.2	Пример настройки BFD с BGP	142
7.23	Настройка политики маршрутизации PBR	144
7.23.1	Настройка Route-map для BGP	144
7.23.2	Route-map на основе списков доступа (Policy-based routing)	149
7.24	Настройка GRE-туннелей	152
7.24.1	Алгоритм настройки.....	152
7.24.2	Пример настройки IP-GRE-туннеля	154
7.25	Настройка L2TPv3-туннелей.....	156
7.25.1	Алгоритм настройки.....	156
7.25.2	Пример настройки L2TPv3-туннеля	158
7.26	Настройка IPsec VPN	159
7.26.1	Настройка Route-based IPsec VPN	159
7.26.2	Настройка Policy-based IPsec VPN.....	168
7.27	Настройка LT-туннелей	176
7.27.1	Алгоритм настройки.....	176
7.27.2	Пример настройки.....	177
7.28	Настройка удаленного доступа к корпоративной сети по PPTP-протоколу.....	178
7.28.1	Алгоритм настройки.....	178
7.28.2	Пример настройки PPTP-сервера.....	180
7.29	Настройка удаленного доступа к корпоративной сети по L2TP over IPsec протоколу	182
7.29.1	Алгоритм настройки.....	182
7.29.2	Пример настройки.....	184
7.30	Настройка удаленного доступа к корпоративной сети по OpenVPN протоколу.....	186
7.30.1	Алгоритм настройки.....	186
7.30.2	Пример настройки.....	188
7.31	Настройка клиента удаленного доступа по протоколу PPPoE.....	190
7.31.1	Алгоритм настройки.....	190
7.31.2	Пример настройки PPPoE-клиента.....	191
7.32	Настройка клиента удаленного доступа по протоколу PPTP	192
7.32.1	Алгоритм настройки.....	192
7.32.2	Пример настройки удаленного подключения по PPTP-протоколу	193
7.33	Настройка клиента удаленного доступа по протоколу L2TP	194
7.33.1	Пример настройки удаленного подключения по L2TP-протоколу	196
7.34	Настройка Dual-Homing	197
7.34.1	Алгоритм настройки.....	197

7.34.2	Пример настройки	197
7.35	Настройка QoS	198
7.35.1	Базовый QoS	198
7.35.2	Расширенный QoS	202
7.36	Настройка зеркалирования	207
7.36.1	Алгоритм настройки.....	207
7.36.2	Пример настройки	208
7.37	Настройка Netflow.....	209
7.37.1	Алгоритм настройки.....	209
7.37.2	Пример настройки	210
7.38	Настройка sFlow.....	210
7.38.1	Алгоритм настройки.....	211
7.38.2	Пример настройки	211
7.39	Настройка LACP.....	212
7.39.1	Алгоритм настройки.....	212
7.39.2	Пример настройки	213
7.40	Настройка VRRP	214
7.40.1	Алгоритм настройки.....	214
7.40.2	Пример настройки 1	217
7.40.3	Пример настройки 2	218
7.41	Настройка VRRP tracking	220
7.41.1	Алгоритм настройки.....	220
7.41.2	Пример настройки	222
7.42	Настройка VRF Lite.....	224
7.42.1	Алгоритм настройки.....	224
7.42.2	Пример настройки	225
7.43	Настройка MultiWAN.....	226
7.43.1	Алгоритм настройки.....	226
7.43.2	Пример настройки	229
7.44	Настройка SNMP	230
7.44.1	Алгоритм настройки.....	231
7.44.2	Пример настройки	233
7.45	Настройка BRAS (Broadband Remote Access Server).....	234
7.45.1	Алгоритм настройки.....	234
7.45.2	Пример настройки BRAS с использованием ПО SoftWLC	238
7.45.3	Пример настройки BRAS без использования ПО SoftWLC.....	244
7.46	Настройка VoIP	249
7.46.1	Процесс настройки SIP-профиля.....	249
7.46.2	Процесс настройки FXS/FXO-портов	250
7.46.3	Процесс настройки плана нумерации.....	251
7.46.4	Пример настройки VoIP	251
7.46.5	Пример настройки плана нумерации	254
8	ЧАСТО ЗАДАВАЕМЫЕ ВОПРОСЫ	258

1 ВВЕДЕНИЕ

1.1 Аннотация

В настоящее время осуществляются масштабные проекты по построению сетей связи. Одной из основных задач при реализации крупных мультисервисных сетей является создание надежных и высокопроизводительных транспортных сетей, которые являются опорными в многослойной архитектуре сетей следующего поколения.

Маршрутизаторы серии RTT-R могут использоваться на сетях крупных предприятий и предприятий малого и среднего бизнеса (SMB), в операторских сетях. Устройства обеспечивают высокую производительность, высокую пропускную способность и поддерживают функции защиты передаваемых данных.

В данном руководстве по эксплуатации изложены назначение, технические характеристики, функции, конструктивное исполнение, порядок установки, рекомендации по начальной настройке и обновлению программного обеспечения маршрутизатора серии RTT-R (далее устройство).

1.2 Целевая аудитория

Данное руководство пользователя предназначено для технического персонала, выполняющего установку, настройку и мониторинг устройств посредством интерфейса командной строки (CLI), а также процедуры по обслуживанию системы и обновлению ПО. Квалификация технического персонала предполагает знание основ работы стеков протоколов TCP/IP, принципов построения Ethernet-сетей.

1.3 Условные обозначения

Обозначение	Описание
<i>Курсив Calibri</i>	Курсивом Calibri указываются переменные или параметры, которые необходимо заменить соответствующим словом или строкой.
Полужирный курсив	Полужирным шрифтом выделены примечания и предупреждения.
<Полужирный курсив>	В угловых скобках указываются названия клавиш на клавиатуре.
Courier New	Полужирным Шрифтом Courier New записаны примеры ввода команд.
Courier New	Шрифтом Courier New в рамке с тенью указаны результаты выполнения команд.
[]	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.
« »	Данный знак в описании команды обозначает «или».

Примечания и предупреждения



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



Предупреждения информируют пользователя о ситуациях, которые могут нанести вред устройству или человеку, привести к некорректной работе устройства или потере данных.

2 ОПИСАНИЕ ИЗДЕЛИЯ

2.1 Назначение

Устройства серии RTT-R являются высокопроизводительными многоцелевыми сетевыми маршрутизаторами. Устройство объединяет в себе традиционные сетевые функции и комплексный многоуровневый подход к безопасности маршрутизации, что позволяет обеспечить надежную защиту для корпоративной среды.

Устройство поддерживает функции межсетевого экрана для защиты своей сетевой инфраструктуры и сочетает в себе новейшие средства обеспечения безопасности данных, шифрования, аутентификации и защиты от вторжений.

Устройство содержит в себе средства для программной и аппаратной обработки данных. За счет оптимального распределения функций обработки данных между частями достигается максимальная производительность.

2.2 Функции

2.2.1 Функции интерфейсов

В таблице 1 приведен список функций интерфейсов устройства.

Таблица 1 – Функции интерфейсов устройства

Определение полярности подключения кабеля (Auto MDI/MDIX)	Автоматическое определение типа кабеля - перекрестный кабель или кабель прямого подключения. <ul style="list-style-type: none">– MDI (Medium Dependent Interface – прямой) – стандарт кабелей для подключения оконечных устройств;– MDIX (Medium Dependent Interface with Crossover – перекрестный) – стандарт кабелей для подключения концентраторов и коммутаторов.
Поддержка обратного давления (Back pressure)	Метод обратного давления используется на полудуплексных соединениях для регулирования потока данных от встречного устройства путем создания коллизий. Метод позволяет избежать переполнения буферной памяти устройства и потери данных.
Управление потоком (IEEE 802.3X)	Управление потоком позволяет соединять низкоскоростное устройство с высокоскоростным. Для предотвращения переполнения буфера низкоскоростное устройство имеет возможность отправлять пакет PAUSE, тем самым информируя высокоскоростное устройство о необходимости сделать паузу при передаче пакетов.
Агрегирование каналов (LAG, Link aggregation)	Агрегирование (объединение) каналов позволяет увеличить пропускную способность канала связи и повысить его надежность. Маршрутизатор поддерживает статическое и динамическое агрегирование каналов. При динамическом агрегировании используется протокол LACP для управления группой каналов.

2.2.2 Функции при работе с MAC-адресами

В таблице 2 приведены функции устройства при работе с MAC-адресами.

Таблица 2 – Функции работы с MAC-адресами

Таблица MAC-адресов	Таблица MAC-адресов устанавливает соответствие между MAC-адресами и интерфейсами устройства и используется для маршрутизации пакетов данных. Маршрутизаторы имеют таблицу емкостью до 16K MAC-адресов и резервируют определенные MAC-адреса для использования системой.
Режим обучения	MAC-таблица может содержать либо статические адреса, либо адреса, изученные при прохождении пакетов данных через устройство. Изучение происходит за счет регистрации MAC-адресов отправителей пакетов с привязкой их к портам и VLAN. Впоследствии эти данные используются для маршрутизации встречных пакетов. Время хранения зарегистрированных MAC-адресов ограничено, его продолжительность может настраиваться администратором. Если MAC-адрес получателя, указанный в принятом устройством пакете, отсутствует в таблице, то такой пакет отправляется далее, как широковещательный в пределах L2 сегмента сети.

2.2.3 Функции второго уровня сетевой модели OSI

В таблице 3 приведены функции и особенности второго уровня (уровень 2 OSI).

Таблица 3 – Описание функций второго уровня (уровень 2 OSI)

Поддержка VLAN	VLAN (Virtual Local Area Network) – это средство разделения сети на изолированные сегменты на уровне L2. Использование VLAN позволяет повысить устойчивость работы крупных сетей за счет деления их на более мелкие сети, изолировать разнородный трафик данных между собой и решить многие другие задачи. Маршрутизаторы поддерживают различные способы организации VLAN: <ul style="list-style-type: none"> – VLAN на базе меток пакетов данных, в соответствии с IEEE 802.1Q; – VLAN на базе портов устройства (port-based); – VLAN на базе использования правил классификации данных (policy-based).
Протокол связующего дерева (Spanning Tree Protocol) ¹	Задачей протокола Spanning Tree является исключение избыточных сетевых соединений и приведение топологии сети к древовидной. Основные применения протокола связаны с предотвращением зацикливания сетевого трафика и с организацией резервных каналов связи.

¹ В текущей версии ПО данный функционал поддерживается только на маршрутизаторе R-800

2.2.4 Функции третьего уровня сетевой модели OSI

В таблице 4 приведены функции третьего уровня (уровень 3 OSI).

Таблица 4 – Описание функций третьего уровня (Layer 3)

Статические IP-маршруты	Администратор маршрутизатора имеет возможность добавлять и удалять статические записи в таблицу маршрутизации.
Динамическая маршрутизация	Протоколы динамической маршрутизации позволяют устройству обмениваться маршрутной информацией с соседними маршрутизаторами и автоматически составлять таблицу маршрутов. Маршрутизатор поддерживает следующие протоколы: RIP, OSPFv2, OSPFv3, BGP.
Таблица ARP	ARP (Address Resolution Protocol) – протокол для выяснения соответствия адресов сетевого и канального уровней. Таблица ARP содержит информацию об изученном соответствии. Соответствие устанавливается на основе анализа ответов от сетевых устройств, адреса устройств запрашиваются с помощью широковещательных пакетов.
Клиент DHCP	Протокол DHCP (Dynamic Host Configuration Protocol) даёт возможность автоматизировать управление сетевыми устройствами. Клиент DHCP позволяет маршрутизатору получать сетевой адрес и дополнительные параметры от внешнего DHCP-сервера. Как правило, этот способ используется для получения сетевых настроек оператора публичной сети (WAN).
Сервер DHCP	Сервер DHCP предназначен для автоматизации и централизации конфигурирования сетевых устройств. Размещение DHCP-сервера на маршрутизаторе позволяет получить законченное решение для поддержки локальной сети. DHCP-сервер, входящий в состав маршрутизатора, позволяет назначать IP-адреса сетевым устройствам и передавать дополнительные сетевые параметры – адреса серверов, адреса шлюзов сети и другие необходимые параметры.
Трансляция сетевых адресов (NAT, Network Address Translation)	Трансляция сетевых адресов – это механизм, который позволяет преобразовывать IP-адреса и номера портов транзитных пакетов. Функция NAT позволяет использовать меньшее количество IP-адресов, транслируя несколько IP-адресов внутренней сети в один внешний публичный IP-адрес. Использование NAT позволяет увеличить защищённость локальной сети за счёт скрытия её внутренней структуры. Маршрутизаторы поддерживают следующие варианты NAT: <ul style="list-style-type: none"> – Source NAT (SNAT) – выполняется замена адреса, а также номера порта источника при прохождении пакета в одну сторону и обратной замене адреса назначения в ответном пакете; – Destination NAT (DNAT) – когда обращения извне транслируются межсетевым экраном на компьютер пользователя в локальной сети, имеющий внутренний адрес и потому недоступный извне сети непосредственно (без NAT).

2.2.5 Функции туннелирования трафика

Таблица 5 – Функции туннелирования трафика

Протоколы туннелирования	<p>Туннелирование – это способ преобразования пакетов данных при передаче их по сети, при котором происходит замена, модификация или добавление нового сетевого заголовка пакета. Такой способ может быть использован для согласования транспортных протоколов при прохождении данных через транзитную сеть, для создания защищенных соединений, при которых туннелированные данные подвергаются шифрованию.</p> <p>Маршрутизаторы поддерживают следующие виды туннелей:</p> <ul style="list-style-type: none"> – GRE - инкапсуляция IP-пакета в другой IP-пакет с добавлением GRE (General Routing Encapsulation) заголовка; – IPv4-IPv4 – туннель, использующий инкапсуляцию исходных IP-пакетов в IP-пакеты с другими сетевыми параметрами; – L2TPv3 – туннель для передачи L2-трафика с помощью IP-пакетов; – IPsec – туннель с шифрованием передаваемых данных; – L2TP, PPTP – туннели, использующиеся для организации удаленного доступа клиент-сервер.
---------------------------------	--

2.2.6 Функции управления и конфигурирования

Таблица 6 – Основные функции управления и конфигурирования

Загрузка и выгрузка файла настройки	Параметры устройства сохраняются в файле настройки, который содержит данные конфигурации как всей системы в целом, так и определенного порта устройства. Для передачи файлов могут использоваться протоколы TFTP, FTP, SCP.
Интерфейс командной строки (CLI)	Управление посредством CLI осуществляется локально через последовательный порт RS-232 либо удаленно через Telnet, SSH. Интерфейс командной строки консоли (CLI) является промышленным стандартом. Интерпретатор CLI предоставляет список команд и ключевых слов для помощи пользователю и сокращению объема вводимых данных.
Syslog	Протокол Syslog обеспечивает передачу информационных сообщений о происходящих в системе событиях и ведение журнала событий.
Сетевые утилиты ping, traceroute	Утилиты ping и traceroute – предназначены для проверки доступности сетевых устройств и для определения маршрутов передачи данных в IP-сетях.
Управление контролируемым доступом – уровни привилегий	Маршрутизаторы поддерживают управление уровнем доступа пользователей к системе. Уровни доступа позволяют управлять зонами ответственности администраторов устройств. Уровни доступа нумеруются от 1 до 15, уровень 15 соответствует полному доступу к управлению устройством.
Аутентификация	<p>Аутентификация – это процедура проверки подлинности пользователя. Маршрутизаторы поддерживают следующие методы аутентификации:</p> <ul style="list-style-type: none"> – локальная – для аутентификации используется локальная база данных пользователей, хранящаяся на самом устройстве; – групповая – база данных пользователей хранится на сервере аутентификации. Для взаимодействия с сервером используются протоколы RADIUS и TACACS.
Сервер SSH Сервер Telnet	Функции сервера SSH и Telnet позволяют установить соединение с устройством для управления им.
Автоматическое восстановление конфигурации	Устройство поддерживает автоматическую систему восстановления конфигурации, которая предотвращает ситуации потери удаленного доступа к устройству после смены конфигурации. Если в течение заданного времени после изменения конфигурации не было введено подтверждение – произойдет автоматический откат конфигурации до предыдущего

2.2.7 Функции сетевой защиты

В таблице приведены функции сетевой защиты, выполняемые устройством.

Таблица 7 – Функции сетевой защиты

Зоны безопасности	Все интерфейсы маршрутизатора распределяются по зонам безопасности. Для каждой пары зон настраиваются правила, определяющие возможность или невозможность прохождения данных между зонами, правила фильтрации трафика данных.
Фильтрация данных	Для каждой пары зон безопасности составляется набор правил, которые позволяют управлять фильтрацией данных, проходящих через маршрутизатор. Командный интерфейс устройства предоставляет средства для детальной настройки правил классификации трафика и для назначения результирующего решения о пропуске трафика.

2.3 Основные технические характеристики

Основные технические параметры маршрутизатора приведены в таблице 8.

Таблица 8 – Основные технические характеристики

Общие параметры		
Пакетный процессор	RTT-R800	Broadcom XLP316L
	RTT-R200	Broadcom XLP204
	RTT-R100	Broadcom XLP104
Интерфейсы	RTT-R800	24 x Ethernet 10/100/1000Base-T 2 x 10GBase-R/1000Base-X (SFP+/SFP)
	RTT-R200	4 x Ethernet 10/100/1000Base-T / 1000Base-X Combo 4 x Ethernet 10/100/1000Base-T
	RTT-R100	4 x Ethernet 10/100/1000Base-T / 1000Base-X Combo
Типы оптических трансиверов	RTT-R800	1000BASE-X SFP, 10GBASE-R SFP+
	RTT-R100	1000BASE-X SFP
	RTT-R200	1000BASE-X SFP
Дуплексный и полудуплексный режимы интерфейсов		- дуплексный и полудуплексный режим для электрических портов - дуплексный режим для оптических портов
Максимальная пропускная способность маршрутизатора (при аппаратной коммутации)	RTT-R800	88 Гбит/с

Скорость передачи данных	RTT-R800	- электрические интерфейсы 10/100/1000 Мбит/с - оптические интерфейсы 1/10 Гбит/с
	RTT-R100 RTT-R200	- электрические интерфейсы 10/100/1000 Мбит/с - оптические интерфейсы 1 Гбит/с
Таблица MAC-адресов	RTT-R800	16K записей
	RTT-R100 RTT-R200	2K записей на бридж
Поддержка VLAN		до 4K активных VLAN в соответствии с 802.1Q
Количество интерфейсов L3	RTT-R800 RTT-R200 RTT-R100	2000
Количество маршрутов BGP	RTT-R800	2,6M
	RTT-R100 RTT-R200	1,2M
Количество маршрутов OSPF	RTT-R800	500K
	RTT-R100 RTT-R200	300K
Количество маршрутов RIP		10K
Количество статических маршрутов		11K
Размер FIB	RTT-R800	1,7M
	RTT-R100 RTT-R200	550K
Соответствие стандартам		IEEE 802.3 10BASE-T Ethernet IEEE 802.3u 100BASE-T Fast Ethernet IEEE 802.3ab 1000BASE-T Gigabit Ethernet IEEE 802.3z Fiber Gigabit Ethernet ANSI/IEEE 802.3 автоопределение скорости IEEE 802.3x контроль потоков данных IEEE 802.3ad объединение каналов LACP IEEE 802.1Q виртуальные локальные сети VLAN IEEE 802.1v IEEE 802.3ac IEEE 802.3ae IEEE 802.1D IEEE 802.1w IEEE 802.1s
Управление		
Локальное управление		CLI
Удаленное управление		TELNET, SSH
Физические характеристики и условия окружающей среды		
Источники питания	RTT-R800	сеть переменного тока: 220В+-20%, 50 Гц сеть постоянного тока: -36 .. - 72В варианты питания: - один источник питания постоянного или переменного тока; - два источника питания постоянного или переменного тока, с возможностью горячей замены.
	RTT-R100 RTT-R200	сеть переменного тока: 220В+-20%, 50 Гц

Максимально потребляемая мощность	RTT-R800	75 Вт
	RTT-R100	20 Вт
	RTT-R200	25 Вт
Масса	RTT-R800	не более 3,6 кг
	RTT-R100 RTT-R200	не более 2,5 кг
Габаритные размеры	RTT-R800	430x352x44 мм
	RTT-R100 RTT-R200	310x240x44 мм
Интервал рабочих температур	RTT-R800 RTT-R100 RTT-R200	от -10 до +45 °С
Интервал температуры хранения		от -40 до +70 °С
Относительная влажность при эксплуатации (без образования конденсата)		не более 80%
Относительная влажность при хранении (без образования конденсата)		от 10% до 95%
Средний срок службы		20 лет

2.4 Конструктивное исполнение

В данном разделе описано конструктивное исполнение устройства. Представлены изображения передней, задней и боковых панелей устройства. Описаны разъемы, светодиодные индикаторы и органы управления.

Устройство выполнено в металлическом корпусе с возможностью установки в 19" конструктив, высота корпуса 1U.

2.4.1 Конструктивное исполнение RTT-R800

2.4.1.1 Передняя панель устройства RTT-R800

Внешний вид передней панели показан на рисунке 1.

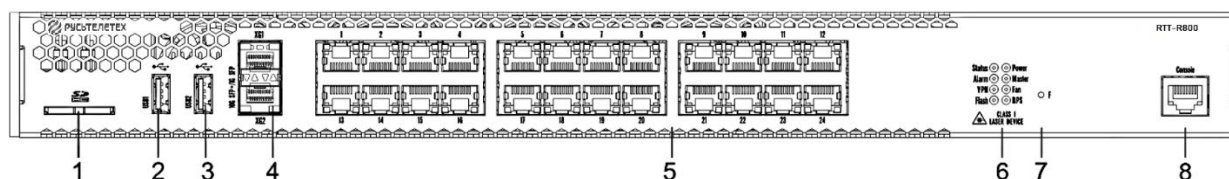


Рисунок 1– Передняя панель RTT-R800

В таблице 9 приведен перечень разъемов, светодиодных индикаторов и органов управления, расположенных на передней панели устройства RTT-800.

Таблица 9 – Описание разъемов, индикаторов и органов управления передней панели RTT-R800

№	Элемент панели передней	Описание
1	SD	Разъем для установки SD-карт памяти.
2	USB1	Порт для подключения USB-устройств.
3	USB2	Порт для подключения USB-устройств.
4	XG1, XG2	Слоты для установки трансиверов 10G SFP+/ 1G SFP.
5	[1 .. 24]	24 порта Gigabit Ethernet 10/100/1000 Base-T (RJ-45).
6	Status	Индикатор текущего состояния устройства.
	Alarm	Индикатор наличия и уровня аварии устройства.
	VPN	Индикатор наличия активных VPN-сессий.
	Flash	Индикатор активности обмена с накопителем данных - SD-картой или USB Flash.
	Power	Индикатор питания устройства.
	Master	Индикатор работы устройства в failover-режимах.
	Fan	Индикатор аварии вентиляторов.
	RPS	Индикатор резервного источника электропитания.
7	F	Функциональная кнопка для перезагрузки устройства и сброса к заводским настройкам: <ul style="list-style-type: none"> – при удержании кнопки менее 10 секунд происходит перезагрузка устройства; – при удержании кнопки более 10 секунд происходит перезагрузка устройства и сброс к заводским настройкам.
8	Console	Консольный порт RS-232 для локального управления устройством.

2.4.1.2 Задняя панель устройств RTT-800

Внешний вид задней панели устройств RTT-R800 приведен на рисунке 2¹.

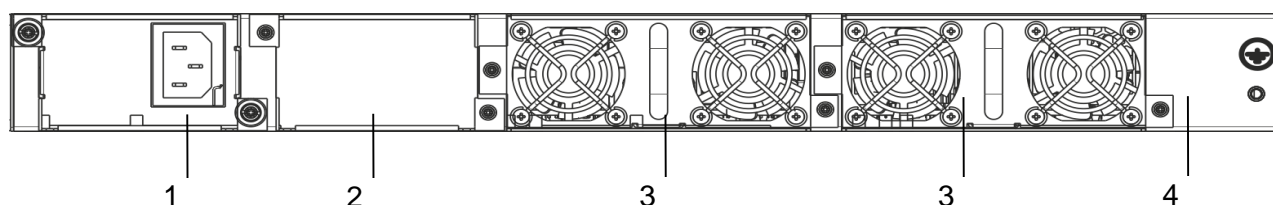


Рисунок 2 – Задняя панель RTT-R800

В таблице приведен перечень разъемов, расположенных на задней панели маршрутизатора.

¹ На рисунке показана комплектация маршрутизатора с одним источником питания переменного тока.

Таблица 10 – Описание разъемов задней панели маршрутизатора

№	Описание
1	Основной источник питания.
2	Место для установки резервного источника питания.
3	Съемные вентиляционные модули с возможностью горячей замены.
4	Клемма для заземления устройства.

2.4.1.3 Боковые панели устройства

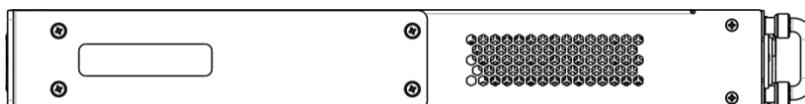


Рисунок 3 – Правая боковая панель маршрутизаторов RTT-R800

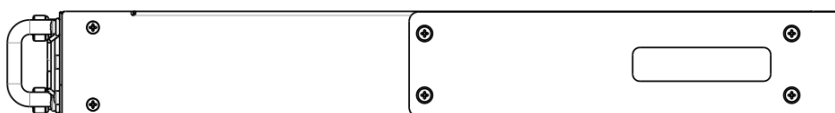


Рисунок 4 – Левая боковая панель маршрутизаторов RTT-R800

На боковых панелях устройства расположены вентиляционные решетки, которые служат для отвода тепла. Не закрывайте вентиляционные отверстия посторонними предметами. Это может привести к перегреву компонентов устройства и вызвать нарушения в его работе. Рекомендации по установке устройства расположены в разделе «Установка и подключение».

2.4.2 Конструктивное исполнение RTT-R100, RTT-R200

2.4.2.1 Передняя панель устройств RTT-R100, RTT-R200

Внешний вид передней панели RTT-R100 показан на рисунке 5.

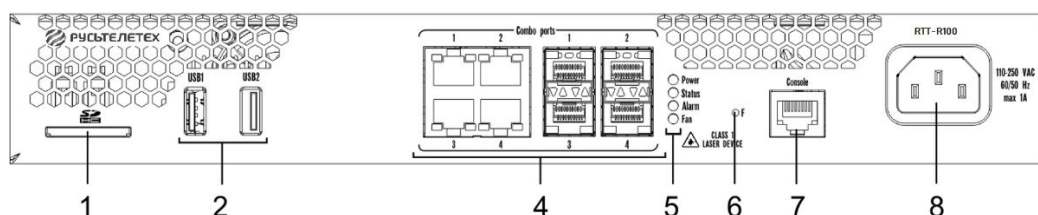


Рисунок 5 – Передняя панель RTT-R100

Внешний вид передней панели RTT-R200 показан на рисунке 6.

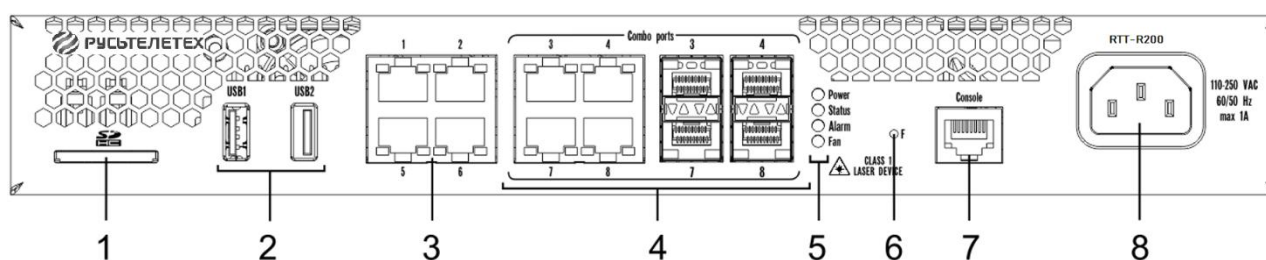


Рисунок 6 – Передняя панель RTT-R200

В таблице 11 приведен перечень разъемов, светодиодных индикаторов и органов управления, расположенных на передней панели устройств RTT-R100, RTT-R200.

Таблица 11 – Описание разъемов, индикаторов и органов управления передней панели

№	Элемент панели передней	Описание
1	SD	Разъем для установки SD-карт памяти.
2	USB1, USB2	2 порта для подключения USB-устройств.
3	[1 .. 4]	4 порта Gigabit Ethernet 10/100/1000 Base-T (RJ-45).
4	Combo Ports	4 порта Gigabit Ethernet 10/100/1000 Base-X (SFP).
5	Power	Индикатор питания устройства.
	Status	Индикатор текущего состояния устройства.
	Alarm	Индикатор наличия и уровня аварии устройства.
	Fan	Индикатор аварии вентиляторов.
6	F	Функциональная кнопка для перезагрузки устройства и сброса к заводским настройкам: <ul style="list-style-type: none"> – при удержании кнопки менее 10 секунд происходит перезагрузка устройства; – при удержании кнопки более 10 секунд происходит перезагрузка устройства и сброс к заводским настройкам.
7	Console	Консольный порт RS-232 для локального управления устройством.
8	110-250 VAC 60/50 Hz max 1A	Источник питания.

2.4.2.2 Задняя панель устройств RTT-R100, RTT-R200

Внешний вид задней панели устройств RTT-R100, RTT-R200 приведен на рисунке 7.

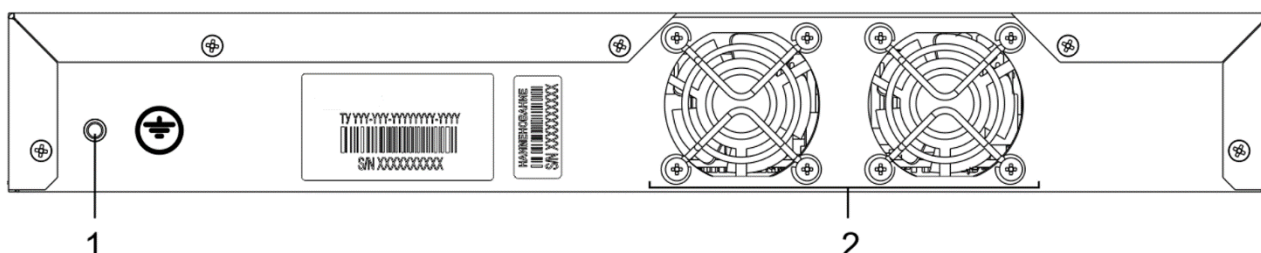


Рисунок 7 – RTT-R100, RTT-R200 задняя панель

В таблице 12 приведен перечень разъемов, расположенных на задней панели маршрутизатора.

Таблица 12 – Описание разъемов задней панели маршрутизатора

№	Описание
1	Клемма для заземления устройства.
2	Вентиляционный модуль.

2.4.2.3 Боковые панели устройства RTT-R100, RTT-R200

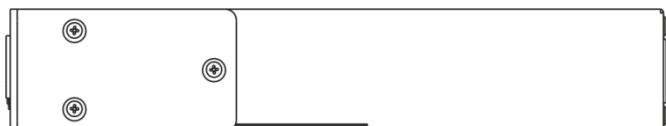


Рисунок 8 – Правая боковая панель маршрутизатора RTT-R100, RTT-R200



Рисунок 9 – Левая боковая панель маршрутизатора RTT-R100

2.4.3 Световая индикация

2.4.3.1 Световая индикация RTT-R800

Состояние медных интерфейсов GigabitEthernet отображается двумя светодиодными индикаторами - *LINK/ACT* зеленого цвета и *SPEED* янтарного цвета. Расположение индикаторов медных интерфейсов показано на рисунке 10. Состояние SFP-интерфейсов отображается двумя индикаторами *RX/ACT* и *TX/ACT* и указано на рисунке 11. Значения световой индикации описаны в таблицах 13 и 14.

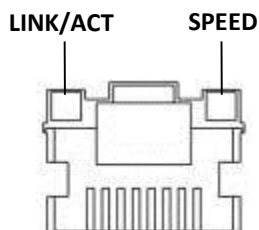
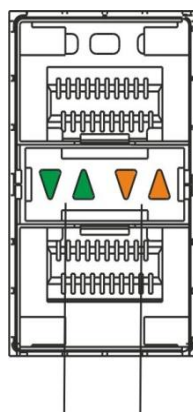


Рисунок 10 – Расположение индикаторов разъема RJ-45



RX/ACT TX/ACT

Рисунок 11 – Расположение индикаторов оптических интерфейсов

Таблица 13 – Световая индикация состояния медных интерфейсов

Свечение индикатора SPEED	Свечение индикатора LINK/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Выключен	Горит постоянно	Установлено соединение на скорости 10 или 100 Мбит/с
Горит постоянно	Горит постоянно	Установлено соединение на скорости 1000 Мбит/с
X	Мигание	Идет передача данных

Таблица 14 – Световая индикация состояния SFP/SFP+ интерфейсов

Свечение индикатора RX/ACT	Свечение индикатора TX/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Горит постоянно	Горит постоянно	Соединение установлено
Мигание	X	Идет прием данных
X	Мигание	Идет передача данных

В следующей таблице приведено описание состояний системных индикаторов устройства и их значений.

Таблица 15 – Состояния системных индикаторов

Название индикатора	Функция индикатора	Состояние индикатора	Состояние устройства
Status	Индикатор текущего состояния устройства.	Зеленый	Устройство работает нормально
		Оранжевый	Устройство находится в состоянии загрузки ПО
Alarm	Индикатор наличия и уровня аварии устройства.	-	-
VPN	Индикатор наличия активных VPN-сессий.	-	-
Flash	Индикатор активности обмена с	Оранжевый	Выполнение операций

	накопителем данных: SD-картой или USB Flash.		чтения/записи по команде «сору»
Power	Индикатор питания устройства.	Зеленый	Питание устройства в норме. Основной источник питания, если он установлен, работает нормально
		Оранжевый	Неработоспособность основного источника питания, авария или отсутствие первичной сети
		Выключен	Отказ внутренних источников питания устройства
Master	Индикатор работы устройства в failover-режимах.	-	-
Fan	Состояние вентилятора охлаждения.	Выключен	Все вентиляторы исправны
		Красный	Отказ одного или более вентиляторов. Причиной возникновения аварии может быть неработоспособность хотя бы одного из вентиляторов – остановка или пониженная частота оборотов.
RPS	Режим работы резервного источника питания.	Зеленый	Резервный источник установлен и исправен
		Выключен	Резервный источник не установлен
		Красный	Отсутствие первичного питания резервного источника или его неисправность

2.4.3.2 Световая индикация RTT-R100/RTT-R200

Состояние медных интерфейсов GigabitEthernet и SFP-интерфейсов отображается двумя светодиодными индикаторами - *LINK/ACT* зеленого цвета и *SPEED* янтарного цвета. Расположение индикаторов медных интерфейсов показано на рисунке 12. Значения световой индикации описаны в таблице 16.

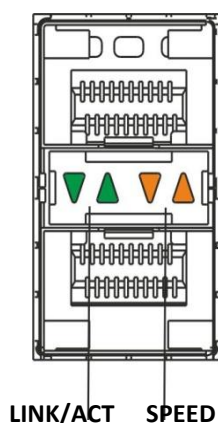


Рисунок 12 – Расположение индикаторов оптических интерфейсов

Таблица 16 – Световая индикация состояния медных интерфейсов и SFP-интерфейсов

Свечение индикатора SPEED	Свечение индикатора LINK/ACT	Состояние интерфейса Ethernet
Выключен	Выключен	Порт выключен или соединение не установлено
Выключен	Горит постоянно	Установлено соединение на скорости 10

		или 100 Мбит/с
Горит постоянно	Горит постоянно	Установлено соединение на скорости 1000 Мбит/с
X	Мигание	Идет передача данных

В следующей таблице приведено описание состояний системных индикаторов устройства и их значений.

Таблица 17 – Состояния системных индикаторов

Название индикатора	Функция индикатора	Состояние индикатора	Состояние устройства
<i>Status</i>	Индикатор текущего состояния устройства.	Зеленый	Устройство работает нормально.
		Оранжевый	Устройство находится в состоянии загрузки ПО.
<i>Alarm</i>	Индикатор наличия и уровня аварии устройства ¹ .	-	-
<i>Power</i>	Индикатор питания устройства.	Зеленый	Питание устройства в норме. Основной источник питания, если он установлен, работает нормально.
		Оранжевый	Неработоспособность основного источника питания, авария или отсутствие первичной сети.
		Выключен	Отказ внутренних источников питания устройства.
<i>Fan</i>	Состояние вентилятора охлаждения.	Выключен	Все вентиляторы исправны.
		Красный	Отказ одного или более вентиляторов. Причиной возникновения аварии может быть неработоспособность хотя бы одного из вентиляторов – остановка или пониженная частота оборотов.

2.5 Комплект поставки

В базовый комплект поставки RTT-R100 входят:

- маршрутизатор RTT-R100;
- кабель питания;
- кабель для подключения к порту Console (RJ-45 – DB9F);
- комплект для крепления устройства в стойку 19”;
- документация.

В базовый комплект поставки RTT-R200 входят:

- маршрутизатор RTT-R200;
- кабель питания;

¹ Не поддерживается в текущей версии ПО

- кабель для подключения к порту Console (RJ-45 – DB9F);
- комплект для крепления устройства в стойку 19”;
- документация.

В базовый комплект поставки RTT-R800 входят:

- маршрутизатор RTT-R800;
- кабель питания;
- кабель для подключения к порту Console (RJ-45 – DB9F);
- комплект для крепления устройства в стойку 19”;
- документация.



По заказу покупателя для RTT-R800 в комплект поставки может быть включен модуль питания (PM160-220/12 или PM75-48/12).



По заказу покупателя в комплект поставки могут быть включены SFP/SFP+-трансиверы.

3 УСТАНОВКА И ПОДКЛЮЧЕНИЕ

В данном разделе описаны процедуры установки устройства в стойку и подключения к питающей сети.

3.1 Крепление кронштейнов

В комплект поставки устройства входят кронштейны для установки в стойку и винты для крепления кронштейнов к корпусу устройства. Для установки кронштейнов:

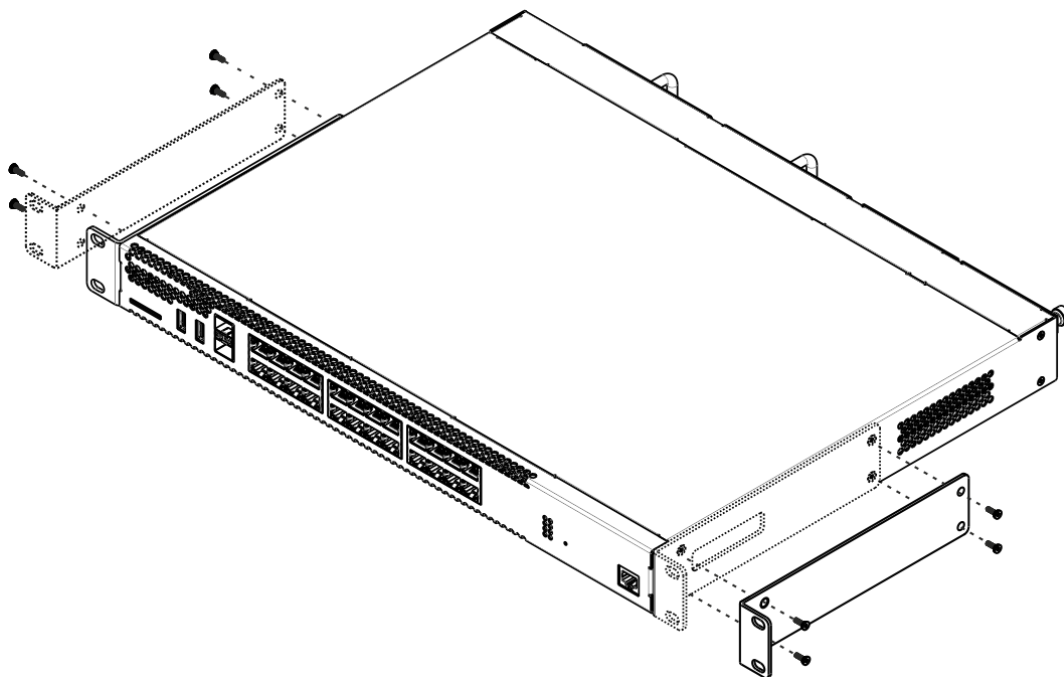


Рисунок 13 – Крепление кронштейнов

1. Совместите четыре отверстия для винтов на кронштейне с такими же отверстиями на боковой панели устройства.
2. С помощью отвертки прикрепите кронштейн винтами к корпусу.
3. Повторите действия 1, 2 для второго кронштейна.

3.2 Установка устройства в стойку

Для установки устройства в стойку:

1. Приложите устройство к вертикальным направляющим стойки.
2. Совместите отверстия кронштейнов с отверстиями на направляющих стойки. Используйте отверстия в направляющих на одном уровне с обеих сторон стойки для того, чтобы устройство располагалось горизонтально.
3. С помощью отвертки прикрепите маршрутизатор к стойке винтами.

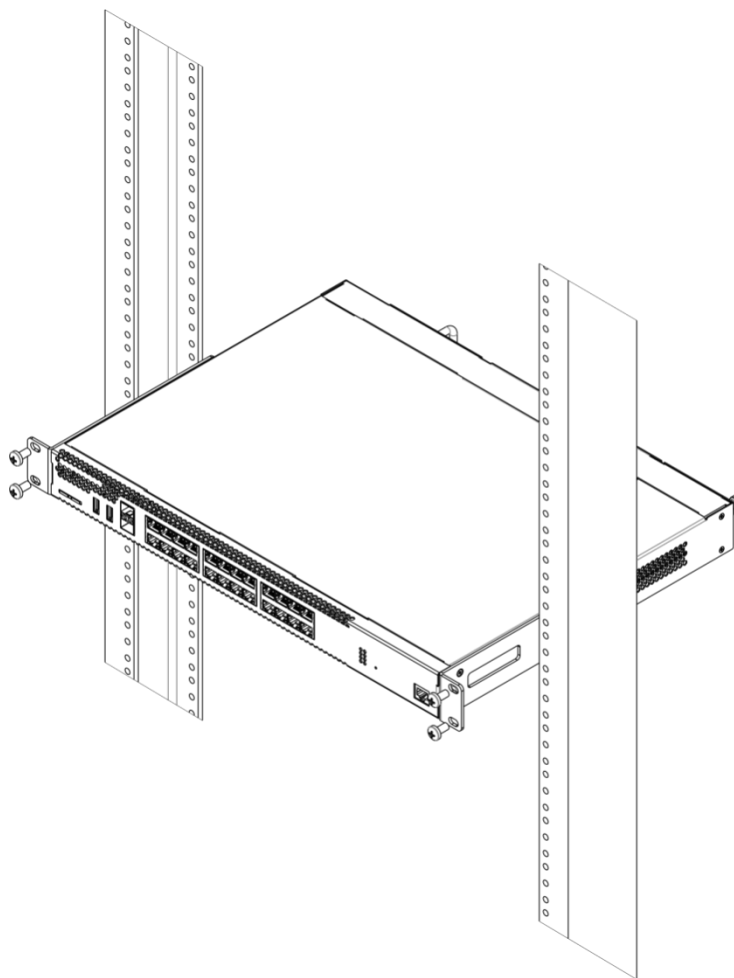


Рисунок 14 – Установка устройства в стойку



Вентиляция устройства организована по схеме фронт-тыл. На передней и боковых панелях устройства расположены вентиляционные отверстия, с задней стороны устройства расположены вентиляционные модули. Не закрывайте входные и выходные вентиляционные отверстия посторонними предметами во избежание перегрева компонентов устройства и нарушения его работы.

3.3 Установка модулей питания RTT-R800

Маршрутизаторы RTT-R800 могут работать с одним или двумя модулями питания. Установка второго модуля питания необходима в случае использования устройства в условиях, требующих повышенной надежности.

Места для установки модулей питания с электрической точки зрения равноценны. С точки зрения использования устройства, модуль питания, находящийся ближе к краю, считается основным, ближе к центру – резервным. Модули питания могут устанавливаться и извлекаться без выключения устройства. При установке или извлечении дополнительного модуля питания маршрутизатор продолжает работу без перезапуска.

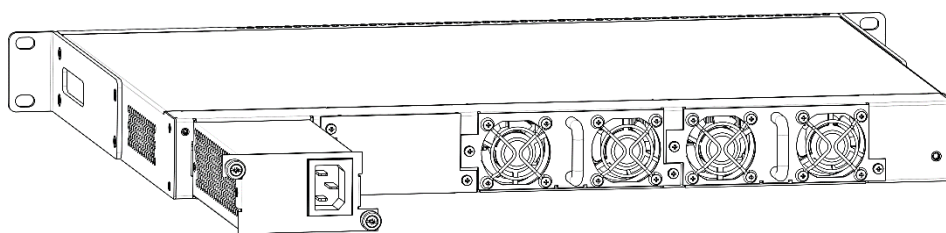


Рисунок 15 – Установка модулей питания

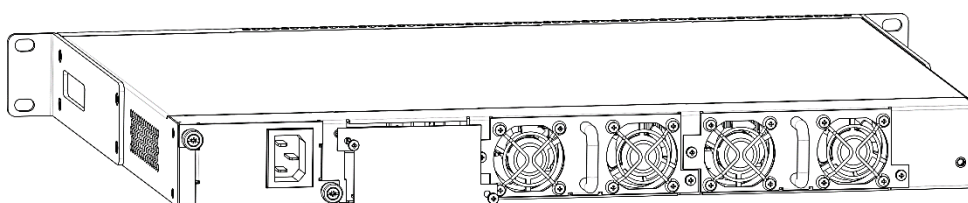


Рисунок 16 – Установка заглушки



Индикация аварии модуля питания может быть вызвана не только отказом модуля, но и отсутствием первичного питания.

Состояние модулей питания может быть проверено по индикации на передней панели маршрутизатора или по диагностике, доступной через интерфейсы управления маршрутизатором.

3.4 Подключение питающей сети

1. Прежде, чем к устройству будет подключена питающая сеть, необходимо заземлить корпус устройства. Заземление необходимо выполнять изолированным многожильным проводом. Устройство заземления и сечение заземляющего провода должны соответствовать требованиями Правил устройства электроустановок (ПУЭ).
2. Если предполагается подключение компьютера или иного оборудования к консольному порту маршрутизатора, это оборудование также должно быть надежно заземлено.
3. Подключите к устройству кабель питания. В зависимости от комплектации устройства, питание может осуществляться от сети переменного тока либо от сети постоянного тока. При подключении сети переменного тока следует использовать кабель,

входящий в комплект устройства. Для подключения к сети постоянного тока используйте провод сечением не менее 1 мм².

4. Включите питание устройства и убедитесь в отсутствии аварий по состоянию индикаторов на передней панели.

3.5 Установка и удаление SFP-трансиверов



Установка оптических модулей может производиться как при выключенном, так и при включенном устройстве.

3.5.1 Установка трансивера

1. Вставьте верхний SFP-модуль в слот открытой частью разъема вниз, а нижний SFP-модуль - открытой частью разъема вверх.

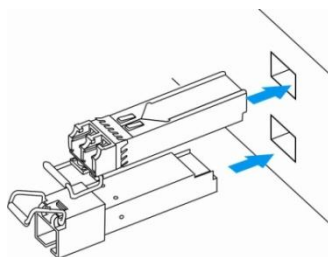


Рисунок 17 – Установка SFP-трансиверов

2. Надавите на модуль по направлению внутрь корпуса устройства до появления характерного щелчка фиксации модуля.

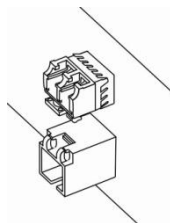


Рисунок 18 – Установленные SFP-трансиверы

3.5.2 Удаление трансивера

1. Откиньте рукоятку модуля, это приведет к разблокированию удерживающей защелки.

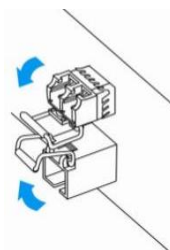


Рисунок 19 – Открытие защелки SFP-трансиверов

2. Извлеките модуль из слота.

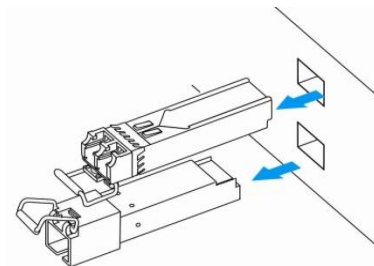


Рисунок 20 – Извлечение SFP-трансиверов

4 ИНТЕРФЕЙСЫ УПРАВЛЕНИЯ

Настройка и мониторинг устройства может осуществляться через различные интерфейсы управления.

Для доступа к устройству может использоваться сетевое подключение по протоколам Telnet и SSH или прямое подключение через консольный порт, соответствующий спецификации RS-232. При доступе по протоколам Telnet, SSH и при подключении через консольный порт для управления устройством используется интерфейс командной строки.



Заводская конфигурация содержит описание доверенной зоны `trusted` и IP-адрес для доступа к управлению устройством - `192.168.1.1/24`.

В доверенную зону входят интерфейсы:

для RTT-R100: GigabitEthernet 1/0/2-4;

для RTT-R200: GigabitEthernet 1/0/2-8;

для RTT-R800: GigabitEthernet 1/0/2-24;

В заводской конфигурации по умолчанию создан пользователь «admin» с паролем «password».

При использовании любого из перечисленных интерфейсов управления действуют единые принципы работы с конфигурацией. Должна соблюдаться определенная, описанная здесь, последовательность изменения и применения конфигурации, позволяющая защитить устройство от некорректного конфигурирования.

4.1 Интерфейс командной строки (CLI)

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 протоколу Telnet, SSH или прямое подключение через консольный порт (например, HyperTerminal).

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

В системе может быть создано необходимое количество пользователей, права доступа задаются индивидуально для каждого из них.


Для обеспечения безопасности командного интерфейса, все команды разделены на две категории – привилегированные и непривилегированные. К привилегированным в основном относятся команды конфигурирования. К непривилегированным – команды мониторинга.

Система позволяет нескольким пользователям одновременно подключаться к устройству.

4.2 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 18 – Типы и порядок именования интерфейсов маршрутизатора

Тип интерфейса	Обозначение
Физические интерфейсы	<p>Обозначение физического интерфейса включает в себя его тип и идентификатор.</p> <p>Идентификатор физических интерфейсов имеет вид <UNIT>/<SLOT>/<PORT>, где</p> <ul style="list-style-type: none"> - <UNIT> – номер устройства в группе устройств, - <SLOT> – номер модуля в составе устройства или 0 при отсутствии деления устройства на модули, - <PORT> – порядковый номер порта.
Порты 1Гбит/с	<p>gigabitethernet <UNIT>/<SLOT>/<PORT></p> <p>Пример обозначения: gigabitethernet 1/0/12</p> <p>Примечание: Допускается использовать сокращенное наименование, например gi1/0/12.</p>
Порты 10Гбит/с	<p>tengigabitethernet <UNIT>/<SLOT>/<PORT></p> <p>Пример обозначения: tengigabitethernet 1/0/2</p> <p>Примечание: Допускается использовать сокращенное наименование, например te1/0/2.</p>
Группы агрегации каналов	<p>Обозначение группы агрегации каналов включает в себя его тип и порядковый номер интерфейса:</p> <p>port-channel <CHANNEL_ID></p> <p>Пример обозначения: port-channel 6</p> <p> Допускается использовать сокращенное наименование, например, po1.</p>
Субинтерфейсы	<p>Обозначение субинтерфейса образуется из обозначения базового интерфейса и идентификатора (VLAN) субинтерфейса, разделенных точкой.</p> <p>Примеры обозначений:</p> <p>gigabitethernet 1/0/12.100</p> <p>tengigabitethernet 1/0/2.123</p> <p>port-channel 1.6</p> <p>Примечание: Идентификатор субинтерфейса может принимать значения [1..4094].</p>
Q-in-Q интерфейсы	<p>Обозначение Q-in-Q интерфейса образуется из обозначения базового интерфейса, идентификатора сервисного VLAN и идентификатора пользовательского VLAN, разделенных точкой.</p> <p>Примеры обозначений:</p> <p>gigabitethernet 1/0/12.100.10</p> <p>tengigabitethernet 1/0/2.45.12</p> <p>port-channel 1.6.34</p> <p>Примечание: Идентификатор сервисного и пользовательского VLAN может принимать значения [1..4094].</p>
E1-интерфейсы	<p>Обозначение E1-интерфейса включает в себя его тип и идентификатор.</p> <p>Идентификатор E1-интерфейсов имеет вид <UNIT>/<SLOT>/<STREAM>, где</p> <ul style="list-style-type: none"> - <UNIT> – номер устройства в группе устройств, - <SLOT> – номер E1-модуля в составе устройства, - <STREAM> – порядковый номер E1-потока. <p>Пример обозначения: e1 1/0/1</p>

Группы агрегации E1-каналов	Обозначение группы агрегации E1-каналов включает в себя его тип и порядковый номер интерфейса: multilink <CHANNEL_ID> Пример обозначения: multilink <CHANNEL_ID>
Логические интерфейсы	Обозначение логического интерфейса является порядковым номером интерфейса: Примеры обозначений: loopback 4 bridge 60 service-port 1



1. Количество интерфейсов каждого типа зависит от модели маршрутизатора.
2. Текущая версия ПО не поддерживает стекирование устройств. Номер устройства в группе устройств unit может принимать только значение 1.
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 gi1/0/1-3,gi1/0/7,te1/0/1

4.3 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 19 – Типы и порядок именования туннелей маршрутизатора

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
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



Количество туннелей каждого типа зависит от модели и ПО маршрутизатора.

5 НАЧАЛЬНАЯ НАСТРОЙКА МАРШРУТИЗАТОРА

5.1 Заводская конфигурация маршрутизатора RTT

При отгрузке устройства потребителю на устройство загружена заводская конфигурация, которая включает минимально необходимые базовые настройки. Заводская конфигурация позволяет использовать маршрутизатор в качестве шлюза с функцией SNAT без необходимости применять дополнительные настройки. Кроме того, заводская конфигурация содержит настройки, позволяющие получить сетевой доступ к устройству для выполнения расширенного конфигурирования.

5.1.1 Описание заводской конфигурации

Для подключения к сетям в конфигурации описаны 2 зоны безопасности с наименованиями «Trusted» для локальной сети и «Untrusted» для публичной сети. Все интерфейсы разделены между двух зон безопасности:

1. Зона «Untrusted» предназначена для подключения к публичной сети (WAN). В этой зоне открыты порты DHCP-протокола для получения динамического IP-адреса от провайдера. Все входящие соединения из данной зоны на маршрутизатор запрещены.

В данную зону безопасности входят интерфейсы:

для RTT-R100/200: GigabitEthernet 1/0/1;

для RTT-R800: GigabitEthernet1/0/1, TengigabitEthernet1/0/1, TengigabitEthernet1/0/2.

Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 2*.

2. Зона «Trusted» предназначена для подключения к локальной сети (LAN). В этой зоне открыты порты протоколов Telnet и SSH для удаленного доступа, ICMP-протокола для проверки доступности маршрутизатора, DHCP-протокола для получения клиентами IP-адресов от маршрутизатора. Исходящие соединения из данной зоны в зону «Untrusted» разрешены.

В данную зону безопасности входят интерфейсы:

для RTT-R100: GigabitEthernet 1/0/2-4;

для RTT-R200: GigabitEthernet1/0/2-8;

для RTT-R800: GigabitEthernet1/0/2-24;

Интерфейсы зоны объединены в один L2-сегмент через сетевой мост *Bridge 1*.

На интерфейсе *Bridge 2* включен DHCP-клиент для получения динамического IP-адреса от провайдера. На интерфейсе *Bridge 1* сконфигурирован статический IP-адрес 192.168.1.1/24. Созданный IP-интерфейс выступает в качестве шлюза для клиентов локальной сети. Для клиентов локальной сети настроен DHCP пул адресов 192.168.1.2-192.168.1.254 с маской 255.255.255.0. Для получения клиентами локальной сети доступа к Internet на маршрутизаторе включен сервис Source NAT.

Политики зон безопасности настроены следующим образом:

Таблица 20 – Описание политик зон безопасности

Зона, из которой идет трафик	Зона, в которую идет трафик	Тип трафика	Действие
Trusted	Untrusted	TCP, UDP, ICMP	разрешен
Trusted	Trusted	TCP, UDP, ICMP	разрешен
Trusted	self	TCP/23(Telnet), TCP/22(SSH), ICMP, UDP/67(DHCP Server), UDP/123(NTP)	разрешен
Untrusted	self	UDP/68(DHCP Client),	разрешен



Для обеспечения возможности конфигурирования устройства при первом включении в конфигурации маршрутизатора создана учётная запись администратора 'admin'. Настоятельно рекомендуется изменить пароль администратора при начальном конфигурировании маршрутизатора.



Для сетевого доступа к управлению маршрутизатором при первом включении в конфигурации задан статический IP-адрес на интерфейсе *Bridge 1* - 192.168.1.1/24.

5.2 Подключение и конфигурирование маршрутизатора

Маршрутизаторы серии RTT-R предназначены для выполнения функций пограничного шлюза и обеспечения безопасности сети пользователя при подключении ее к публичным сетям передачи данных.

Базовая настройка маршрутизатора должна включать:

- назначение IP-адресов (статических или динамических) интерфейсам, участвующим в маршрутизации данных;
- создание зон безопасности и распределение интерфейсов по зонам;
- создание политик, регулирующих прохождение данных между зонами;
- настройка сервисов, сопутствующих маршрутизации данных (NAT, Firewall и прочие).

Расширенные настройки зависят от требований конкретной схемы применения устройства и легко могут быть добавлены или изменены с помощью имеющихся интерфейсов управления.

5.2.1 Подключение к маршрутизатору

Предусмотрены следующие способы подключения к устройству:

5.2.1.1 Подключение по локальной сети Ethernet



При первоначальном старте маршрутизатор загружается с заводской конфигурацией. Описание заводской конфигурации приведено в разделе 5.1 Заводская конфигурация маршрутизатора данного руководства.

Подключите сетевой кабель передачи данных (патч-корд) к любому порту, входящему в зону «**Trusted**», и к компьютеру, предназначенному для управления.

В заводской конфигурации маршрутизатора активирован DHCP-сервер с пулом IP-адресов в подсети **192.168.1.0/24**.

При подключении сетевого интерфейса управляющего компьютера он должен получить сетевой адрес от сервера.

Если IP-адрес не получен по какой-либо причине, то следует назначить адрес интерфейса вручную, используя любой адрес, кроме 192.168.1.1, в подсети 192.168.1.0/24.

5.2.1.2 Подключение через консольный порт RS-232

При помощи кабеля RJ-45/DBF9, который входит в комплект поставки устройства, соедините порт **«Console»** маршрутизатора с портом RS-232 компьютера.

Запустите терминальную программу (например, HyperTerminal или Minicom) и создайте новое подключение. Должен быть использован режим эмуляции терминала VT100.

Выполните следующие настройки интерфейса RS-232:

Скорость: 115200 бит/с
Биты данных: 8 бит
Четность: нет
Стоповые биты: 1
Управление потоком: нет

5.2.2 Применение изменения конфигурации

Любые изменения, внесенные в конфигурацию, вступят в действие только после применения команды:

```
RTT# commit  
Configuration has been successfully committed
```

После применения данной команды запускается таймер "отката" конфигурации. Для остановки таймера и механизма "отката" используется команда:

```
RTT# confirm  
Configuration has been successfully confirmed
```

Значение таймера "отката" по умолчанию – 600 секунд. Для изменения данного таймера используется команда:

```
RTT(config)# system config-confirm timeout <TIME>
```

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

5.2.3 Базовая настройка маршрутизатора

Процедура настройки маршрутизатора при первом включении состоит из следующих этапов:

- Изменение пароля пользователя «admin».
- Создание новых пользователей.
- Назначение имени устройства (Hostname).

- Установка параметров подключения к публичной сети в соответствии с требованиями провайдера.
- Настройка удаленного доступа к маршрутизатору.
- Применение базовых настроек.

5.2.3.1 Изменение пароля пользователя «admin»

Для защищенного входа в систему необходимо сменить пароль привилегированного пользователя «admin».



Учетная запись techsupport (до версии 1.0.7 - rustel) необходима для удаленного обслуживания сервисным центром;

Учетная запись remote - аутентификация RADIUS, TACACS+, LDAP;

Удалить пользователей admin, techsupport, remote нельзя. Можно только сменить пароль и уровень привилегий.

Имя пользователя и пароль вводится при входе в систему во время сеансов администрирования устройства.

Для изменения пароля пользователя «admin» используются следующие команды:

```
RTT# configure
RTT(config)# username admin
RTT(config-user)# password <new-password>
RTT(config-user)# exit
```

5.2.3.2 Создание новых пользователей

Для создания нового пользователя системы или настройки любого из параметров – имени пользователя, пароля, уровня привилегий, – используются команды:

```
RTT(config)# username <name>
RTT(config-user)# password <password>
RTT(config-user)# privilege <privilege>
RTT(config-user)# exit
```



Уровни привилегий 1-9 разрешают доступ к устройству и просмотр его оперативного состояния, но запрещают настройку. Уровни привилегий 10-14 разрешают как доступ, так и настройку большей части функций устройства. Уровень привилегий 15 разрешает как доступ, так и настройку всех функций устройства.

Пример команд для создания пользователя «fedor» с паролем «12345678» и уровнем привилегий 15 и создания пользователя «ivan» с паролем «password» и уровнем привилегий 1:

```
RTT# configure
RTT(config)# username fedor
RTT(config-user)# password 12345678
RTT(config-user)# privilege 15
RTT(config-user)# exit
RTT(config)# username ivan
RTT(config-user)# password password
RTT(config-user)# privilege 1
RTT(config-user)# exit
```

5.2.3.3 Назначение имени устройства

Для назначения имени устройства используются следующие команды:

```
RTT# configure
RTT(config)# hostname <new-name>
```

После применения конфигурации приглашение командной строки изменится на значение, заданное параметром **<new-name>**.

5.2.3.4 Настройка параметров публичной сети

Для настройки сетевого интерфейса маршрутизатора в публичной сети необходимо назначить устройству параметры, определённые провайдером сети - IP-адрес, маска подсети и адрес шлюза по умолчанию.

Пример команд настройки статического IP-адреса для субинтерфейса **GigabitEthernet 1/0/2.150** для доступа к маршрутизатору через **VLAN 150**.

Параметры интерфейса:

- IP-адрес – 192.168.16.144;
- Маска подсети – 255.255.255.0;
- IP-адрес шлюза по умолчанию – 192.168.16.1.

```
RTT# configure
RTT(config)# interface gigabitethernet 1/0/2.150
RTT(config-subif)# ip address 192.168.16.144/24
RTT(config-subif)# exit
RTT(config)# ip route 0.0.0.0/0 192.168.16.1
```

Для того чтобы убедиться, что адрес был назначен интерфейсу, после применения конфигурации введите следующую команду:

```
RTT# show ip interfaces
```

IP address	Interface	Type
-----	-----	-----
192.168.16.144/24	gigabitethernet 1/0/2.150	static

Провайдер может использовать динамически назначаемые адреса в своей сети. Для получения IP-адреса может использоваться протокол DHCP, если в сети присутствует сервер DHCP.

Пример настройки, предназначенной для получения динамического IP-адреса от DHCP-сервера на интерфейсе **GigabitEthernet 1/0/10**:

```
RTT# configure
RTT(config)# interface gigabitethernet 1/0/10
RTT(config-if)# ip address dhcp
RTT(config-if)# exit
```

Для того чтобы убедиться, что адрес был назначен интерфейсу, введите следующую команду после применения конфигурации:

```
RTT# show ip interfaces
```

IP address	Interface	Type
-----	-----	-----
192.168.11.5/25	gigabitethernet 1/0/10	DHCP

5.2.3.5 Настройка удаленного доступа к маршрутизатору

В заводской конфигурации разрешен удаленный доступ к маршрутизатору по протоколам Telnet или SSH из зоны **«trusted»**. Для того чтобы разрешить удаленный доступ к маршрутизатору из других зон, например, из публичной сети, необходимо создать соответствующие правила в firewall.

При конфигурировании доступа к маршрутизатору правила создаются для пары зон:

- **source-zone** – зона, из которой будет осуществляться удаленный доступ;
- **self** – зона, в которой находится интерфейс управления маршрутизатором.

Для создания разрешающего правила используются следующие команды:

```
RTT# configure
RTT(config)# security zone-pair <source-zone> self
RTT(config-zone-pair)# rule <number>
RTT(config-zone-rule)# action permit
RTT(config-zone-rule)# match protocol tcp
RTT(config-zone-rule)# match source-address <network object-group>
RTT(config-zone-rule)# match destination-address <network object-group>
RTT(config-zone-rule)# match source-port any
RTT(config-zone-rule)# match destination-port <service object-group>
RTT(config-zone-rule)# enable
RTT(config-zone-rule)# exit
RTT(config-zone-pair)# exit
```

Пример команд для разрешения пользователям из зоны **«untrusted»** с IP-адресами **132.16.0.5-132.16.0.10** подключаться к маршрутизатору с IP-адресом **40.13.1.22** по протоколу SSH:

```
RTT# configure
RTT(config)# object-group network clients
RTT(config-addr-set)# ip address-range 132.16.0.5-132.16.0.10
RTT(config-addr-set)# exit
RTT(config)# object-group network gateway
RTT(config-addr-set)# ip address-range 40.13.1.22
RTT(config-addr-set)# exit
RTT(config)# object-group service ssh
RTT(config-port-set)# port-range 22
RTT(config-port-set)# exit
RTT(config)# security zone-pair untrusted self
RTT(config-zone-pair)# rule 10
RTT(config-zone-rule)# action permit
RTT(config-zone-rule)# match protocol tcp
RTT(config-zone-rule)# match source-address clients
RTT(config-zone-rule)# match destination-address gateway
RTT(config-zone-rule)# match source-port any
RTT(config-zone-rule)# match destination-port ssh
RTT(config-zone-rule)# enable
RTT(config-zone-rule)# exit
RTT(config-zone-pair)# exit
```

6 ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

6.1 Обновление программного обеспечения средствами системы



Для обновления программного обеспечения понадобится один из следующих серверов: TFTP, FTP, SCP. На сервер должны быть помещены файлы программного обеспечения маршрутизатора, полученные от производителя.

На маршрутизаторе хранится две копии программного обеспечения. Для обеспечения надежности процедуры обновления программного обеспечения доступна для обновления только копия, которая не была использована для последнего старта устройства.



При обновлении программного обеспечения конфигурация маршрутизатора конвертируется в соответствии с новой версией.

При загрузке маршрутизатора с более старой версией программного обеспечения, чем загруженная ранее, конфигурация не конвертируется и впоследствии удаляется.



Обновление средствами системы доступно в версии 1.0.3.69 и последующих. Обновление ПО с более ранних версий можно произвести, воспользовавшись инструкцией, приведенной в разделе 6.2.

Обновление программного обеспечения на устройстве, работающем под управлением операционной системы, выполняется в следующем порядке.

1. Подготовьте для работы выбранный сервер. Должен быть известен адрес сервера, на сервере должен быть размещен файл дистрибутивный файл программного обеспечения.
2. Маршрутизатор должен быть подготовлен к работе в соответствии с требованиями документации. Конфигурация маршрутизатора должна позволять обмениваться данными по протоколам TFTP/FTP/SCP и ICMP с сервером. При этом должна быть учтена принадлежность сервера к зонам безопасности маршрутизатора.
3. Подключитесь к маршрутизатору локально через консольный порт Console или удаленно, используя проколы Telnet или SSH.

Проверьте доступность сервера для маршрутизатора, используя команду *ping* на маршрутизаторе. Если сервер не доступен – проверьте правильность настроек маршрутизатора и состояние сетевых интерфейсов сервера.

4. Для обновления программного обеспечения маршрутизатора введите следующую команду. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла программного обеспечения, помещенного на сервер (при использовании SCP нужно указать полный путь – параметр *<folder>*). После ввода команды маршрутизатор скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.

TFTP:

```
RTT# copy tftp://<server>:<file_name> system:firmware
```

FTP:

```
RTT# copy ftp://[<user>[:<password>]@]<server>:<file_name>
system:firmware
```

SCP:

```
RTT# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>
system:firmware
```

Для примера обновим основное ПО через SCP:

```
RTT# copy scp://adm:password123@192.168.16.168://home/tftp/firmware
system:firmware
```

- Для того чтобы устройство работало под управлением новой версии программного обеспечения, необходимо произвести переключение активного образа. С помощью команды *show bootvar* следует выяснить номер образа, содержащего обновленное ПО.

```
RTT# show bootvar
```

Image	Version	Date	Status	After
reboot				
1	1.0.4 build 141[f812808]	date 18/02/2015 time 16:12:54	Active	*
2	1.0.4 build 141[f812808]	date 18/02/2015 time 16:12:54	Not Active	

Для выбора образа используйте команду

```
RTT# boot system image-[1|2]
```

- Для обновления вторичного загрузчика (U-Boot) введите следующую команду. В качестве параметра *<server>* должен быть указан IP-адрес используемого сервера. Для обновления с FTP или SCP-сервера потребуется ввести имя пользователя (параметр *<user>*) и пароль (параметр *<password>*). В качестве параметра *<file_name>* укажите имя файла вторичного загрузчика, помещенного на сервер (при использовании SCP нужно указать полный путь – параметр *<folder>*). После ввода команды маршрутизатор скопирует файл во внутреннюю память, проверит целостность данных и сохранит его в энергонезависимую память устройства.

TFTP:

```
RTT# copy tftp://<server>:<file_name> system:boot
```

FTP:

```
RTT# copy ftp://<server>:<file_name> system:boot
```

SCP:

```
RTT# copy scp://[<user>[:<password>]@]<server>://<folder>/<file_name>
system:boot
```

6.2 Обновление программного обеспечения из начального загрузчика

Программное обеспечение маршрутизатора можно обновить из начального загрузчика следующим образом:

1. Остановите загрузку устройства после окончания инициализации маршрутизатора загрузчиком U-Boot, нажав клавишу <Esc>.

```
Configuring PoE...
distribution 1 dest_threshold 0xa drop_timer 0x0
Configuring POE in bypass mode
NAE configuration done!
initializing port 0, type 2.
initializing port 1, type 2.
SMC Endian Test:b81fb81f
nae-0, nae-1
=====Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.
Hit any key to stop autoboot: 2
```

2. Укажите IP-адрес TFTP-сервера:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv serverip 10.100.100.1
```

3. Укажите IP-адрес маршрутизатора:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv ipaddr 10.100.100.2
```

4. Можно сохранить окружение командой «saveenv» для будущих обновлений.
5. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run tftp_update_image1
BRCM.XLP316Lite Rev B0.u-boot# run set_bootpart_1
```

```
Using nae-0-3 device
TFTP from server 10.100.100.1; our IP address is 10.100.100.2
Filename 'RTT800/firmware'.
Load address: 0xa800000060000000
Loading: TftpStart:TftpTimeoutMsecs = 10000, TftpTimeoutCountMax = 6
#####
#####
#####
#####
#####
done
Bytes transferred = 64453909 (3d77d15 hex)
Device 0: MT29F8G08ABBCAH4 ... is now current device

NAND erase: device 0 offset 0x1440000, size 0x6400000
Bad block table found at page 262080, version 0x01
Bad block table found at page 262016, version 0x01
Erasing at 0x7800000 -- 1895825408% complete..
OK

NAND write: device 0 offset 0x1440000, size 0x6400000
104857600 bytes written: OK
```

6. Запустите загруженное программное обеспечение:

```
BRCM.XLP316Lite Rev B0.u-boot# reset
```

6.3 Обновление вторичного загрузчика (U-Boot)

Вторичный загрузчик занимается инициализацией NAND и маршрутизатора. При обновлении новый файл вторичного загрузчика сохраняется на flash на месте старого.

Для просмотра текущей версии загрузочного файла, работающего на устройстве, введите команду «version» в CLI U-Boot, также версия отображается в процессе загрузки маршрутизатора:

```
BRCM.XLP316Lite Rev B0.u-boot# version  
BRCM.XLP.U-Boot:1.1.0.47 (29/11/2016 - 19:00:24)
```

Процедура обновления ПО:

1. Остановите загрузку устройства после окончания инициализации маршрутизатора загрузчиком U-Boot, нажав клавишу <Esc>.

```
Configuring PoE...  
distribution 1 dest_threshold 0xa drop_timer 0x0  
Configuring POE in bypass mode  
NAE configuration done!  
initializing port 0, type 2.  
initializing port 1, type 2.  
SMC Endian Test:b81fb81f  
nae-0, nae-1  
=====Skip: Load SYS UCORE for old 8xxB1/3xxB0 revision on default.  
Hit any key to stop autoboot: 2
```

2. Укажите IP-адрес TFTP-сервера:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv serverip 10.100.100.1
```

3. Укажите IP-адрес маршрутизатора:

```
BRCM.XLP316Lite Rev B0.u-boot# setenv ipaddr 10.100.100.2
```

4. Можно сохранить окружение командой «saveenv» для будущих обновлений.

5. Запустите процедуру обновления программного обеспечения:

```
BRCM.XLP316Lite Rev B0.u-boot# run upd_uboot или BRCM.XLP316LiteRevB0.u-boot#  
runtftp_update_uboot, в зависимости от версии загрузчика.
```

```
Using nae-1 device  
TFTP from server 10.100.100.1; our IP address is 10.100.100.2  
Filename 'RTT800/u-boot.bin'.  
Load address: 0xa800000078020000  
Loading: #####  
done  
Bytes transferred = 852648 (d02a8 hex)  
SF: Detected MX25L12805D with page size 256, total 16777216 bytes  
16384 KiB MX25L12805D at 0:0 is now current device
```

6. Перезагрузите маршрутизатор:

```
BRCM.XLP316Lite Rev B0.u-boot# reset
```

7 ПРИМЕРЫ НАСТРОЙКИ МАРШРУТИЗАТОРА

7.1 Настройка VLAN

VLAN (Virtual Local Area Network) — логическая («виртуальная») локальная сеть, представляет собой группу устройств, которые взаимодействуют между собой на канальном уровне независимо от их физического местонахождения. Работа VLAN основана на использовании дополнительных полей Ethernet-заголовка согласно стандарту 802.1q. По сути, VLAN изолирует широковещательный домен путем ограничения коммутации Ethernet-фреймов только с одинаковым VLAN-ID в Ethernet-заголовке.

7.1.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать VLAN.	<code>RTT(config)# vlan <VID></code>	<VID> – идентификатор VLAN, задаётся в диапазоне [2..4094]. Также есть возможность создания нескольких vlan (через запятую) или диапазона vlan (через дефис).
2	Задать имя vlan (не обязательно).	<code>RTT(config-vlan)# name <vlan-name></code>	<vlan-name> - до 255 символов.
3	Задать режим работы L2 интерфейса.	<code>RTT(config-if-gi)# switchport access</code>	Только для RTT-100/200.
		<code>RTT(config-if-gi)# switchport trunk</code>	Только для RTT-100/200.
		<code>RTT(config-gi)# switchport general</code>	Только для RTT-R800.
4	Настроить список vlan на интерфейсе в тегированном режиме.	<code>RTT(config-if-gi)# switchport trunk allowed vlan add <VID></code>	Для RTT-100/200. <VID> – идентификатор VLAN, задаётся в диапазоне [2..4094]. Также есть возможность создания нескольких vlan (через запятую) или диапазона vlan (через дефис).
		<code>RTT(config-if-gi)# switchport general allowed vlan add <VID> tagged</code>	Для RTT-R800. <VID> – идентификатор VLAN, задаётся в диапазоне [2..4094]. Также есть возможность создания нескольких vlan (через запятую) или диапазона vlan (через дефис).
5	Настроить список vlan на интерфейсе в нетегированном режиме (не обязательно).	<code>RTT(config-if-gi)# switchport trunk native-vlan <VID></code>	Для RTT-100/200. <VID> – идентификатор VLAN, задаётся в диапазоне [2..4094].
		<code>RTT(config-if-gi)# switchport general allowed vlan add <VID> untagged</code>	Для RTT-R800. <VID> – идентификатор VLAN, задаётся в диапазоне [2..4094].

7.1.2 Пример настройки 1. Удаление VLAN с интерфейса

Задача:

На основе заводской конфигурации удалить из VLAN 2 порт gi1/0/1.



Рисунок 21 – Схема сети

Решение:

Удалим VLAN 2 с порта gi1/0/1:

```
RTT(config)# interface gi 1/0/1
RTT(config-if-gi)# switchport general allowed vlan remove 2 untagged
RTT(config-if-gi)# no switchport general pvid
```

7.1.3 Пример настройки 2. Разрешение обработки VLAN в тегированном режиме

Задача:

Настроить порты gi1/0/1 и gi1/0/2 для передачи и приема пакетов в VLAN 2, VLAN 64, VLAN 2000.

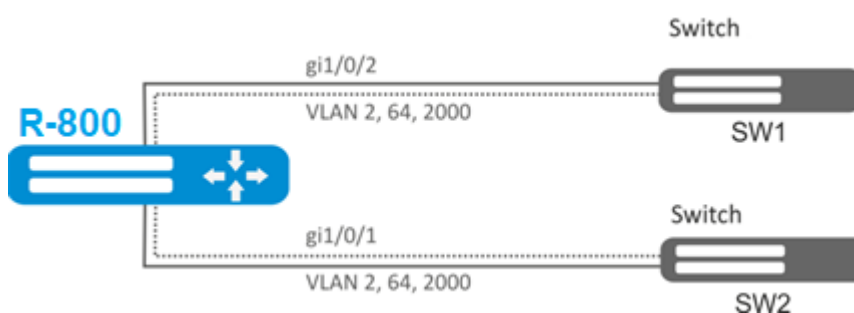


Рисунок 22 – Схема сети

Решение:

Создадим VLAN 2, VLAN 64, VLAN 2000 на RTT-R800:

```
RTT-R800(config)# vlan 2,64,2000
```

Пропишем VLAN 2, VLAN 64, VLAN 2000 на порт gi1/0/1-2:

```
RTT-R800(config)# interface gi1/0/1
RTT-R800(config-if-gi)# switchport forbidden default-vlan
RTT-R800(config-if-gi)# switchport general allowed vlan add 2,64,2000 tagged
```

7.1.4 Пример настройки 3. Разрешение обработки VLAN в тегированном и не тегированном режиме

Задача:

Настроить порты gi1/0/1 для передачи и приема пакетов в VLAN 2, VLAN 64, VLAN 2000 в режиме trunk, настроить порт gi1/0/2 в режиме access для VLAN 2 на RTT-100/RTT-200.

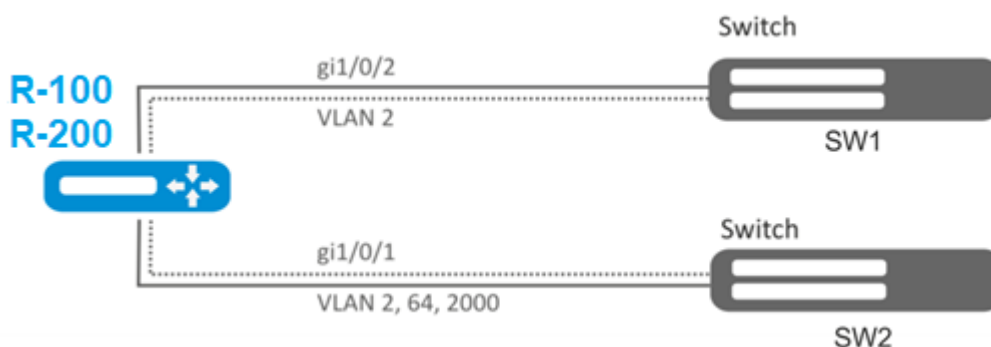


Рисунок 23 – Схема сети

Решение:

Создадим VLAN 2, VLAN 64, VLAN 2000 на RTT-100/ RTT-200:

```
RTT(config)# vlan 2,64,2000
```

Пропишем VLAN 2, VLAN 64, VLAN 2000 на порт gi1/0/1:

```
RTT(config)# interface gi1/0/1
RTT(config-if-gi)# switchport forbidden default-vlan
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 2,64,2000
```

Пропишем VLAN 2 на порт gi1/0/2:

```
RTT(config)# interface gi1/0/2
RTT(config-if-gi)# switchport access vlan 2
```

7.2 Настройка LLDP

Link Layer Discovery Protocol (LLDP) — протокол канального уровня, позволяющий сетевому оборудованию оповещать оборудование, работающее в локальной сети, о своём существовании и передавать ему свои характеристики, а также получать от него аналогичные сведения.

7.2.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать LLDP на маршрутизаторе	RTT(config)# lldp enable	
2	Установить период, в течение которого маршрутизатор хранит информацию, полученную по LLDP (не обязательно)	RTT(config)# lldp hold-multiplier <SEC>	<SEC> – период времени в секундах, принимает значение [1..10].

3	Установить IP-адрес, который будет передаваться в LLDP TLV в качестве management-address (не обязательно)	<code>RTT(config)# lldp management-address <ADDR></code>	<ADDR> – IP-адрес, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. По умолчанию задается один из существующих
4	Установить поле system-description, которое будет передаваться в LLDP TLV в качестве system-description. (не обязательно)	<code>RTT(config)# lldp system-description <DESCRIPTION></code>	<DESCRIPTION> – описание системы, задается строкой до 255 символов. По умолчанию содержит информацию о модели и версии ПО маршрутизатора.
5	Установить поле system-name, которое будет передаваться в LLDP TLV в качестве system-name. (не обязательно)	<code>RTT(config)# lldp system-name <NAME></code>	<NAME> – имя системы, задается строкой до 255 символов. По умолчанию совпадает с заданным hostname
6	Установить период отправки LLDPDU (не обязательно)	<code>RTT(config)# lldp timer <SEC></code>	<SEC> – период времени в секундах, принимает значение [1..32768].
7	Включить прием и обработку LLDPDU на физическом интерфейсе.	<code>RTT(config-if-gi)# lldp receive</code>	
8	Включить отправку LLDPDU на физическом интерфейсе.	<code>RTT(config-if-gi)# lldp transmit</code>	

7.2.2 Пример настройки

Задача:

Организовать обмен и обработку LLDPDU между маршрутизаторами RTT-1 и RTT-2.



Рисунок 24 – Схема сети

Решение:

1. Конфигурирование R1

Включим LLDP глобально на маршрутизаторе:

```
RTT(config)# lldp enable
```

Включим прием и отправку LLDPDU на интерфейсе gi 1/0/1.

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# lldp receive
```

```
RTT(config-if-gi) # lldp transmit
```

2. Конфигурирование R2

Включим LLDP глобально на маршрутизаторе:

```
RTT(config) # lldp enable
```

Включим прием и отправку LLDPDU на интерфейсе gi 1/0/1.

```
RTT(config) # interface gigabitethernet 1/0/1
```

```
RTT(config-if-gi) # lldp receive
```

```
RTT(config-if-gi) # lldp transmit
```

Общую информацию по LLDP соседям можно посмотреть командой:

```
RTT# show lldp neighbors
```

Подробную информацию по соседу конкретного интерфейса можно посмотреть командой:

```
RTT# show lldp neighbors gigabitethernet 1/0/1
```

Общую статистику по LLDP можно посмотреть командой:

```
RTT# show lldp statistics
```

7.3 Настройка LLDP MED

LLDP MED — расширение стандарта LLDP, которое позволяет передавать сетевые политики: VLAN ID, DSCP, priority.

7.3.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать LLDP на маршрутизаторе	RTT(config) # lldp enable	
2	Активировать расширение MED LLDP на маршрутизаторе	RTT(config) # lldp med fast-start enable	
3	Создать сетевую политику	RTT(config) # network-policy <NAME>	<NAME> – имя network-policy, задается строкой до 31 символа.
4	Указать тип приложения	RTT(config-net-policy) # application <APP_TYPE>	<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.

5	Установить значение DSCP	<code>RTT(config-net-policy)# dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].
6	Установить значение COS	<code>RTT(config-net-policy)# priority <PRIORITY></code>	<COS> - значение приоритета, принимает значения: best-effort – COS0; background – COS1; excellente-effort – COS2; critical-applications – COS3; video – COS4; voice – COS5; internetwork-control – COS6; network-control – COS7.
7	Установить значение VLAN ID	<code>RTT(config-net-policy)# vlan <VID> [tagged]</code>	<VID> – идентификационный номер VLAN, принимает значения [1...4094]; tagged – ключ, при установке которого абонентское устройство будет отправлять Ethernet-фреймы указанного приложения в тегированном виде.
8	Установить сетевую политику на интерфейс	<code>RTT(config-if-gi)# lldp network-policy <NAME></code>	<NAME> – имя network-policy, задается строкой до 31 символа.
9	Включить отправку LLDPDU на физическом интерфейсе.	<code>RTT(config-if-gi)# lldp transmit</code>	

7.3.2 Пример настройки Voice VLAN

Voice VLAN — VLAN ID, при получении которого IP-телефон переходит в режим trunk с заданным VLAN ID для приема и отправки VoIP трафика. Передача VLAN ID осуществляется посредством расширения MED протокола LLDP.

Задача:

Необходимо разделить трафик телефонии и данных по разным VLAN, vid 10 для данных и vid 20 для телефонии, и настроить отправку Voice VLAN с порта gi 1/0/1 RTT. При этом на IP-телефоне должен поддерживаться и быть включен Voice VLAN.



Рисунок 25 – Схема сети

Решение:

На примере RTT-R100

Предварительно необходимо создать VLAN 10 и 20 и настроить интерфейс gi 1/0/1 в режиме trunk:

```
RTT(config)# vlan 10,20
RTT(config-vlan)# exit
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# switchport mode trunk
RTT(config-if-gi)# switchport trunk allowed vlan add 10,30
RTT(config-if-gi)# exit
```

Включим LLDP и поддержку MED в LLDP глобально на маршрутизаторе:

```
RTT(config)# lldp enable
RTT(config)# lldp med fast-start enable
```

Создадим и настроим сетевую политику таким образом, чтобы для приложения voice указывался VLAN ID 20:

```
RTT(config)# network-policy VOICE_VLAN
RTT(config-net-policy)# application voice
RTT(config-net-policy)# vlan 20 tagged
RTT(config-net-policy)# exit
```

Настроим LLDP на интерфейсе и установим на него сетевую политику:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# lldp transmit
RTT(config-if-gi)# lldp receive
RTT(config-if-gi)# lldp network-policy VOICE_VLAN
RTT(config-if-gi)# exit
```

7.4 Настройка терминации на саб-интерфейсе

Для терминирования Ethernet-фреймов конкретного VLAN на определенном физическом интерфейсе необходимо создать саб-интерфейс с указанием номера VLAN, фреймы которого будут терминироваться. При создании двух саб-интерфейсов с одинаковыми VLAN, но на разных физических/агрегированных интерфейсах, коммутация Ethernet-фреймов между данными саб-интерфейсами будет невозможна т.к. сегменты за пределами саб-интерфейсов будут являться

отдельными ширококестельными доменами. Для обмена данными между абонентами разных саб-интерфейсов (даже с одинаковым VLAN-ID) будет использоваться маршрутизация, т.е. обмен данными будет происходить на третьем уровне модели OSI.

7.4.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать саб-интерфейс физического интерфейса.	RTT(config)# interface gigabitethernet <PORT>.<S-VLAN> или interface tengigabitethernet <PORT>.<S-VLAN> или interface port-channel <CH>.<S-VLAN>	<PORT> – номер физического интерфейса. <CH> – номер агрегированного интерфейса. <S-VLAN> – идентификатор создаваемого S-VLAN. Если физический интерфейс включен в bridge-group, создать саб-интерфейс будет невозможно.
2	Задать описание саб-интерфейса (не обязательно).	RTT(config-subif) # description <DESCRIPTION>	<DESCRIPTION> – описание интерфейса, задается строкой до 255 символов.
3	Указать экземпляр VRF, в котором будет работать данный саб-интерфейс (не обязательно).	RTT(config-subif) # ip vrf forwarding <VRF>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Установить интервал времени, в течение которого собирается статистика о нагрузке на саб-интерфейс (не обязательно).	RTT(config-subif) # load- average <TIME>	<TIME> – интервал в секундах, принимает значения [5..150].
5	Включить саб-интерфейс bridge-group (не обязательно).	RTT(config-subif) #bridge-group <BRIDGE-ID>	<BRIDGE-ID> – идентификационный номер моста.
6	Установить время жизни IPv4/IPv6 записей в ARP-таблице, изученных на данном интерфейсе (не обязательно).	RTT(config-subif) # ip arp reachable-time <TIME> или ipv6 nd reachable-time <TIME>	<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>.

7.4.2 Пример настройки саб-интерфейса

Задача:

Настроить терминацию подсети 192.168.3.1/24 в VLAN: 828 на физическом интерфейсе gigabitethernet 1/0/1.

Решение:

Создадим саб-интерфейс для VLAN: 828

```
RTT(config)# interface gigabitethernet 1/0/1.828
```

Настроим IP-адрес из необходимой подсети:

```
RTT(config)# interface gigabitethernet 1/0/1.828
RTT(config-subif)# ip address 192.168.3.1/24
RTT(config-subif)# exit
```



Помимо назначения IP-адреса, на саб-интерфейсе необходимо либо отключить firewall, либо настроить соответствующую зону безопасности.

7.5 Настройка терминации на Q-in-Q-интерфейсе

Q-in-Q - технология передачи пакетов с двумя 802.1q тэгами. Данная технология используется для расширения количества используемых VLAN в сети передачи данных. Внутренним тэгом (InnerTag) называется 802.1q заголовок ближе к payload. Так же внутренний тэг называют C-VLAN (Customer VLAN). Внешний тэг (OuterTag) - это 802.1q заголовок, добавленный к изначальному 802.1q пакетом, так же называется S-VLAN (Service VLAN). Использование двойных меток в Ethernet фреймах описывается протоколом 802.1ad.

7.5.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать саб-интерфейс физического интерфейса.	RTT(config)# interface gigabitethernet <PORT>.<S-VLAN> или interface tengigabitethernet <PORT>.<S-VLAN> или interface port-channel <CH>.<S-VLAN>	<PORT> – номер физического интерфейса. <CH> – номер агрегированного интерфейса. <S-VLAN> – идентификатор создаваемого S-VLAN.
2	Создать q-in-q-интерфейс.	RTT(config)# interface gigabitethernet <PORT>.<S-VLAN>.<C-VLAN> или RTT(config)# interface tengigabitethernet <PORT>.<S-VLAN>.<C-VLAN> или RTT(config)# interface port-channel <CH>.<S-VLAN>.<C-VLAN>	<PORT> – номер физического интерфейса. <CH> – номер агрегированного интерфейса. <S-VLAN> – идентификатор создаваемого S-VLAN. <C-VLAN> – идентификатор создаваемого C-VLAN. Если физический или саб-интерфейс включен в bridge-group, создать саб-интерфейс будет невозможно.
3	Задать описание q-in-q-интерфейс (не обязательно).	RTT(config-qinq-if)# description <DESCRIPTION>	<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.

4	Указать экземпляр VRF, в котором будет работать данный q-in-q-интерфейс (не обязательно).	<code>RTT(config-qinq-if) # ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
5	Установить интервал времени, в течение которого собирается статистика о нагрузке на q-in-q-интерфейс (не обязательно).	<code>RTT(config-qinq-if) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150].
6	Включить q-in-q-интерфейс bridge-group (не обязательно).	<code>RTT(config-qinq-if) # bridge-group <BRIDGE-ID></code>	<BRIDGE-ID> – идентификационный номер моста.
7	Установить время жизни IPv4/IPv6 записей в ARP-таблице изученных на данном q-in-q-интерфейсе (не обязательно).	<code>RTT(config-qinq-if) # ip arp reachable-time <TIME></code> или <code>ipv6 nd reachable-time <TIME></code>	<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>.

7.5.2 Пример настройки q-in-q-интерфейса

Задача:

Настроить терминацию подсети 192.168.1.1/24 комбинации C-VLAN: 741, S-VLAN: 828 на физическом интерфейсе gigabitethernet 1/0/1.

Решение:

Создадим саб-интерфейс для S-VLAN: 828

```
RTT(config)# interface gigabitethernet 1/0/1.828
RTT(config-subif)# exit
```

Создадим q-in-q саб-интерфейс для S-VLAN: 741 и настроим IP-адрес из необходимой подсети.

```
RTT(config)# interface gigabitethernet 1/0/1.828.741
RTT(config-qinq-if) # ip address 192.168.1.1/24
RTT(config-qinq-if) # exit
```



Помимо назначения IP-адреса, на q-in-q саб-интерфейсе необходимо либо отключить firewall, либо настроить соответствующую зону безопасности.

7.6 Настройка USB модемов

Использование USB-модемов позволяет организовать дополнительный канал связи для работы маршрутизатора. При подключении USB-модемов возможно использовать USB-концентраторы. Одновременно в системе может быть сконфигурировано до 10-ти USB-модемов.

7.6.1 Алгоритм настройки USB-модемов

Шаг	Описание	Команда	Ключи
1	После подключения USB-модема дождаться, когда система обнаружит подключенное устройство		
2	Определить, какой номер устройства назначен на подключенный USB-модем	<code>RTT# show cellulars status modem</code>	В поле "USB port" будет указан идентификатор подключенного устройства.
3	Создать профиль настроек для USB-модема и перейти в режим конфигурирования профиля	<code>RTT(config)# cellular profile <ID></code>	<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].
4	Задать описание профиля настроек (не обязательно)	<code>RTT(config-cellular-profile)# description <DESCRIPTION></code>	<DESCRIPTION> – описание интерфейса, задаётся строкой до 255 символов.
5	Задать точку доступа мобильной сети	<code>RTT(config-cellular-profile)# apn <NAME></code>	<NAME> – точка доступа мобильной сети, задаётся строкой до 31 символа.
6	Задать имя пользователя мобильной сети (если мобильный оператор требует данное поле)	<code>RTT(config-cellular-profile)# user <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
7	Установить пароля для пользователя мобильной сети (если мобильный оператор требует данное поле)	<code>RTT(config-cellular-profile)# password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F]; <ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.
8	Установить номер дозвона для подключения к мобильной сети	<code>RTT(config-cellular-profile)# number <WORD></code>	<WORD> - номер дозвона для подключения к мобильной сети, задаётся строкой до 15 символов.
9	Задать метод аутентификации пользователя в мобильной сети (не обязательно)	<code>RTT(config-cellular-profile)# allowed-auth <TYPE></code>	<TYPE> - метод аутентификации пользователя в мобильной сети [none, PAP, CHAP, MSCHAP, MSCHAPv2, EAP].
10	Ограничить возможность использования семейств IP-адресов в мобильной сети.	<code>RTT(config-cellular-profile)# ip-version { ipv4 ipv6 }</code>	ipv4 – семейство IPv4; ipv6 – семейство IPv6;
11	Создать USB-модем в конфигурации маршрутизатора и перейти в режим конфигурирования модема	<code>RTT(config)# cellular modem <ID></code>	<ID> – идентификатор USB-модема в системе [1..10].
12	Указать экземпляр VRF, в котором будет работать данный модем (не обязательно).	<code>RTT(config-cellular-modem)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.

13	Задать идентификатор USB-модема, назначенного системой (определен в пункте 2.)	<code>RTT(config-cellular-modem) # device <WORD></code>	<WORD> – идентификатор USB-порта подключенного модема [1..12].
14	Назначить ранее созданный профиль настроек для USB-модема	<code>RTT(config-cellular-modem) # profile <ID></code>	<ID> – идентификатор профиля настроек для USB-модема в системе [1..10].
15	Задать код разблокировки SIM-карты (в случае необходимости)	<code>RTT(config-cellular-modem) # pin <WORD></code>	<WORD> - код разблокировки SIM-карты [4..8]. Возможно использование только цифр.
16	Разрешить использование того или иного режима работы USB-модема (не обязательно)	<code>RTT(config-cellular-modem) # allowed-mode <MODE></code>	<MODE> – допустимый режим работы USB-модема [2g, 3g, 4g]. По умолчанию: разрешены все режимы, поддерживаемые модемом.
17	Задать размер максимального принимаемого пакета (не обязательно)	<code>RTT(config-cellular-modem) # mru { <MRU> }</code>	<MRU> – значение MRU, принимает значения в диапазоне [128..16383].
18	Задать предпочтительный режим работы USB-модема в мобильной сети (не обязательно)	<code>RTT(config-cellular-modem) # preferred-mode { <MODE> }</code>	<MODE> – предпочтительный режим работы USB-модема [2g, 3g, 4g]
19	Активировать USB-модем	<code>RTT(config-cellular-modem) # enable</code>	

7.6.2 Пример настройки

Задача:

Настроить подключение к сети Интернет, используя USB-модем.

Решение:

Для примера разберём подключение к сотовому оператору МТС.

После подключения модема необходимо дождаться, когда система обнаружит устройство. Определим порт устройства, который был назначен на подключённый USB-модем:

```
RTT# show cellular status modem
```

Number device	USB port	Manufacturer	Model	Current state	Interface	Link state
1	1-2	huawei	E3372	Disabled	--	Down

Создадим профиль настроек для USB-модема:

```
RTT(config) # cellular profile 1
```

Зададим APN, который требует провайдер, или иной необходимый адрес. Ниже показан пример подключения к APN МТС:

```
RTT(config-cellular-profile) # apn internet.mts.ru
```

При необходимости задаём имя пользователя, пароль, номер дозвона и метод аутентификации:

```
RTT(config-cellular-profile) # user mts
RTT(config-cellular-profile) # password ascii-text mts
RTT(config-cellular-profile) # number *99#
RTT(config-cellular-profile) # allowed-auth PAP
```

Перейдём к конфигурированию USB-модема и зададим идентификатор, соответствующий порту устройства, который был определён в начале:

```
RTT(config) # cellular modem 1
RTT(config-cellular-modem) # device 1-2
```

Назначим соответствующий профиль настроек и активируем модем:

```
RTT(config-cellular-modem) # profile 1
RTT(config-cellular-modem) # enable
```

7.7 Настройка AAA

AAA (Authentication, Authorization, Accounting) – используется для описания процесса предоставления доступа и контроля над ним.

- Authentication (аутентификация) – сопоставление персоны (запроса) существующей учётной записи в системе безопасности. Осуществляется по логину, паролю.
- Authorization (авторизация, проверка полномочий, проверка уровня доступа) – сопоставление учётной записи в системе и определённых полномочий.
- Accounting (учёт) – слежение за подключением пользователя или внесённым им изменениям.

7.7.1 Алгоритм настройки локальной аутентификации

Шаг	Описание	Команда	Ключи
1	Указать local в качестве метода аутентификации.	RTT(config) # aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]	<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.
2	Указать enable в качестве способа аутентификации повышения привилегий	RTT(config) # aaa authentication enable <NAME><METHOD 1> [<METHOD 2>]	<NAME> – имя списка, задаётся строкой до 31 символа. Способы аутентификации:

	пользователей.	[<METHOD 3>] [<METHOD 4>]	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.
3	Указать способ перебора методов аутентификации в случае отказа (не обязательно).	RTT(config)# aaa authentication mode <MODE>	<MODE> – способы перебора методов: chain - если сервер вернул FAIL , перейти к следующему в цепочке методу аутентификации; break - если сервер вернул FAIL , прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. Значение по умолчанию: chain.
4	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (не обязательно)	RTT(config)# aaa authentication attempts max-fail <COUNT> <TIME>	<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535]; <TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535]. Значение по умолчанию: <COUNT> - 5; <TIME> - 300
5	Включить запрос на смену пароля по умолчанию для пользователя admin (не обязательно)	RTT(config)# security passwords default- expired	
6	Включить режим запрета на использование ранее установленных паролей локальных пользователей (не обязательно)	RTT(config)# security passwords history <COUNT>	<COUNT> – количество паролей сохраняемых в памяти маршрутизатора. Принимает значение в диапазоне [1..15]. Значение по умолчанию: 0
7	Установить время действия пароля локального пользователя (не обязательно)	RTT(config)# security passwords lifetime <TIME>	<TIME> – интервал времени действия пароля в днях. Принимает значение в диапазоне [1..365]. По умолчанию: Время действия пароля локального пользователя неограниченно.
8	Установить ограничение на минимальную длину пароля локального пользователя и ENABLE-пароля (не обязательно)	RTT(config)# security passwords min-length <NUM>	<NUM> – минимальное количество символов в пароле. Принимает значение в диапазоне [8..128]. Значение по умолчанию: 0

9	Установить ограничение на максимальную длину пароля локального пользователя и ENABLE-пароля (не обязательно)	<code>RTT(config)# security passwords max-length <NUM></code>	<NUM> – максимальное количество символов в пароле. Принимает значение в диапазоне [8..128]. Значение по умолчанию: не ограничено.
10	Установить минимальное количество типов символов, которые должны присутствовать в пароле локального пользователя и ENABLE-пароле (не обязательно)	<code>RTT(config)# security passwords symbol-types <COUNT></code>	<COUNT> – минимальное количество типов символов в пароле. Принимает значение в диапазоне [1..4]. Значение по умолчанию: 1
11	Установить минимальное количество строчных букв в пароле локального пользователя и ENABLE-пароле (не обязательно)	<code>RTT(config)# security passwords lower-case <COUNT></code>	<COUNT> – минимальное количество строчных букв в пароле локального пользователя и ENABLE-пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0
12	Установить минимальное количество прописных (заглавных) букв в пароле локального пользователя и ENABLE-пароле (не обязательно)	<code>RTT(config)# security passwords upper-case <COUNT></code>	<COUNT> – минимальное количество прописных (заглавных) букв в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0
13	Установить минимальное количество цифр в пароле локального пользователя и ENABLE-пароле (не обязательно)	<code>RTT(config)# security passwords numeric-count <COUNT></code>	<COUNT> – минимальное количество цифр в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0
14	Установить минимальное количество специальных символов в пароле локального пользователя и ENABLE-пароле (не обязательно)	<code>RTT(config)# security passwords special-case <COUNT></code>	<COUNT> – минимальное количество специальных символов в пароле. Принимает значение в диапазоне [0..128]. Значение по умолчанию: 0
15	Добавить пользователя в локальную базу и перейти в режим настройки параметров пользователя	<code>RTT(config)# username <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
16	Установить пароль пользователя	<code>RTT(config-user)# password { <CLEAR-TEXT> encrypted <HASH_SHA512> }</code>	<CLEAR-TEXT> – пароль, задаётся строкой [8 .. 31] символов, принимает значения [0-9a-fA-F]; <HASH_SHA512> – хеш пароля по алгоритму sha512, задаётся строкой из 110 символов.
17	Установить уровень привилегий пользователя	<code>RTT(config-user)# privilege <PRIV></code>	<PRIV> – необходимый уровень привилегий. Принимает значение [1..15].
18	Перейти в режим конфигурирования соответствующего	<code>RTT(config)# line console</code> или	

	терминала	RTT(config)# line telnet или RTT(config)# line ssh	
19	Активировать список аутентификации входа пользователей в систему	RTT(config-line-ssh)# login authentication <NAME>	<NAME> – имя списка, задаётся строкой до 31 символа.
20	Активировать список аутентификации повышения привилегий пользователей	RTT(config-line-ssh)# enable authentication <NAME>	<NAME> – имя списка, задаётся строкой до 31 символа.
21	Задать интервал, по истечении которого будет разрываться бездействующая сессия	RTT(config-line-ssh)# exec-timeout <SEC>	<SEC> – период времени в минутах, принимает значения [1..65535].

7.7.2 Алгоритм настройки AAA по протоколу RADIUS

Шаг	Описание	Команда	Ключи
1	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов RADIUS-сервера (не обязательно).	RTT(config)# radius-server dscp <DSCP>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63.
2	Задать глобальное значение количества перезапросов к последнему активному RADIUS-серверу (не обязательно).	RTT(config)# radius-server retransmit <COUNT>	<COUNT> – количество перезапросов к RADIUS-серверу, принимает значения [1..10]. Значение по умолчанию: 1.
3	Задать глобальное значение интервала, по истечении которого маршрутизатор считает, что RADIUS-сервер недоступен (не обязательно).	RTT(config)# radius-server timeout <SEC>	<SEC> – период времени в секундах, принимает значения [1..30]. Значение по умолчанию: 3 секунды.
4	Добавить RADIUS-сервер в список используемых серверов и перейти в режим его конфигурирования.	RTT(config)# radius-server host { <IP-ADDR> <IPV6-ADDR> } [vrf <VRF>] RTT(config-radius-server)#	<IP-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] <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
5	Указать количество	aaa authentication attempts max-fail	<COUNT> – количество неудачных

	неудачных попыток аутентификации для блокировки логина пользователя и времени блокировки (не обязательно).	<COUNT> <TIME>	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: <COUNT> - 5; <TIME> - 300
6	Задать пароль для аутентификации на удаленном RADIUS-сервере.	RTT (config-radius-server) # key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
7	Задать приоритет использования удаленного RADIUS-сервера (не обязательно).	RTT (config-radius-server) # priority <PRIORITY>	<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535]. Чем ниже значение, тем приоритетнее сервер. Значение по умолчанию: 1.
8	Задать интервал, по истечении которого маршрутизатор считает, что данный RADIUS-сервер недоступен (не обязательно).	RTT (config-radius-server) # timeout <SEC>	<SEC> – период времени в секундах, принимает значения [1..30]. Значение по умолчанию: используется значение глобального таймера.
9	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых RADIUS-пакетах.	RTT (config-radius-server) # source-address { <ADDR> <IPV6-ADDR> }	<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].
10	Указать radius в качестве метода аутентификации.	RTT (config) # aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]	<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.

11	Указать radius в качестве способа аутентификации повышения привилегий пользователей.	<pre>RTT(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</pre>	<p><NAME> – имя списка строка до 31 символа;</p> <p>default – имя списка по умолчанию.</p> <p><METHOD> – способы аутентификации:</p> <p>enable – аутентификация с помощью enable-паролей;</p> <p>tacacs – аутентификация по протоколу TACACS;</p> <p>radius – аутентификация по протоколу RADIUS;</p> <p>ldap – аутентификация по протоколу LDAP.</p>
12	Указать способ перебора методов аутентификации в случае отказа (не обязательно).	<pre>RTT(config)# aaa authentication mode <MODE></pre>	<p><MODE> – способы перебора методов:</p> <p>chain - если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации;</p> <p>break - если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. Значение по умолчанию: chain.</p>
13	Сконфигурировать radius в списке способов учета сессий пользователей (не обязательно).	<pre>RTT(config)# aaa accounting login start- stop <METHOD 1> [<METHOD 2>]</pre>	<p><METHOD> – способы учета:</p> <p>tacacs – учет сессий по протоколу TACACS;</p> <p>radius – учет сессий по протоколу RADIUS.</p>
14	Перейти в режим конфигурирования соответствующего терминала.	<pre>RTT(config)# line <TYPE></pre>	<p><TYPE> – тип консоли:</p> <p>console – локальная консоль;</p> <p>ssh – защищенная удаленная консоль.</p>
15	Активировать список аутентификации входа пользователей в систему.	<pre>RTT(config-line- console)# login authentication <NAME></pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа. Создано на шаге 8.</p>
16	Активировать список аутентификации повышения привилегий пользователей.	<pre>RTT(config-line- console)# enable authentication <NAME></pre>	<p><NAME> – имя списка, задаётся строкой до 31 символа. Создано на шаге 9.</p>

7.7.3 Алгоритм настройки AAA по протоколу TACACS

Шаг	Описание	Команда	Ключи
1	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов TACACS-сервера (не обязательно).	<code>RTT(config)# tacacs-server dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63.
2	Задать глобальное значение интервала, по истечении которого маршрутизатор считает, что TACACS-сервер недоступен (не обязательно).	<code>RTT(config)# tacacs-server timeout <SEC></code>	<SEC> – период времени в секундах, принимает значения [1..30]. Значение по умолчанию: 3 секунды.
3	Добавить TACACS-сервер в список используемых серверов и перейти в режим его конфигурирования.	<code>RTT(config)# tacacs-server host { <IP-ADDR> <IPV6-ADDR> } [vrf <VRF>]</code> <code>RTT(config-tacacs-server)#</code>	<IP-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] <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
4	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (не обязательно)	<code>aaa authentication attempts max-fail <COUNT> <TIME></code>	<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535]; <TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535]. Значение по умолчанию: <COUNT> - 5; <TIME> - 300
5	Задать пароль для аутентификации на удаленном TACACS-сервере	<code>RTT(config-tacacs-server)# key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
6	Задать номер порта для обмена данными с удаленным TACACS-сервером (не обязательно).	<code>RTT(config-tacacs-server)# port <PORT></code>	<PORT> – номер TCP-порта для обмена данными с удаленным сервером, принимает значения [1..65535]. Значение по умолчанию: 49 для TACACS-сервера.

7	Задать приоритет использования удаленного TACACS сервера (не обязательно).	<code>RTT(config-tacacs-server)# priority <PRIORITY></code>	<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535]. Чем ниже значение, тем приоритетнее сервер. Значение по умолчанию: 1.
8	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых TACACS-пакетах.	<code>RTT(config-radius-tacacs)# source-address { <ADDR> <IPv6-ADDR> }</code>	<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
9	Указать TACACS в качестве способа аутентификации повышения привилегий пользователей.	<code>RTT(config)# aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</code>	<NAME> – имя списка строка до 31 символа; default – имя списка по умолчанию. <METHOD> – способы аутентификации: enable – аутентификация с помощью enable-паролей; tacacs – аутентификация по протоколу TACACS; radius – аутентификация по протоколу RADIUS; ldap – аутентификация по протоколу LDAP.
10	Указать способ перебора методов аутентификации в случае отказа (не обязательно).	<code>RTT(config)# aaa authentication mode <MODE></code>	<MODE> – способы перебора методов: chain – если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации; break – если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. Значение по умолчанию: chain.
11	Сконфигурировать список способов учета команд, введенных в CLI (не обязательно).	<code>RTT(config)# aaa accounting commands stop-only tacacs</code>	
12	Сконфигурировать tacacs в списке способов учета сессий пользователей (не обязательно).	<code>RTT(config)# aaa accounting login start-stop <METHOD 1> [<METHOD 2>]</code>	<METHOD> – способы учета: tacacs – учет сессий по протоколу TACACS; radius – учет сессий по протоколу RADIUS.
13	Перейти в режим конфигурирования соответствующего терминала.	<code>RTT(config)# line <TYPE></code>	<TYPE> – тип консоли: console – локальная консоль; ssh – защищенная удаленная консоль.
14	Активировать список	<code>RTT(config-line-console)# login</code>	<NAME> – имя списка, задаётся

	аутентификации входа пользователей в систему.	authentication <NAME>	строкой до 31 символа. Создано на шаге 7.
15	Активировать список аутентификации повышения привилегий пользователей.	RTT(config-line-console)# enable authentication <NAME>	<NAME> – имя списка, задаётся строкой до 31 символа. Создано на шаге 8.

7.7.4 Алгоритм настройки AAA по протоколу LDAP

Шаг	Описание	Команда	Ключи
1	Задать базовый DN (Distinguished name), который будет использоваться при поиске пользователей.	<code>RTT(config)# ldap-server base-dn <NAME></code>	<NAME> – базовый DN, задается строкой до 255 символов.
2	Задать интервал, по истечении которого устройство считает, что LDAP-сервер недоступен (не обязательно).	<code>RTT(config)# ldap-server bind timeout <SEC></code>	<SEC> – период времени в секундах, принимает значения [1..30]. Значение по умолчанию: 3 секунды.
3	Задать DN (Distinguished name) пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.	<code>RTT(config)# ldap-server bind authenticate root-dn <NAME></code>	<NAME> – DN пользователя с правами администратора, задается строкой до 255 символов.
4	Задать пароль пользователя с правами администратора, под которым будет происходить авторизация на LDAP-сервере при поиске пользователей.	<code>RTT(config)# ldap-server bind authenticate root-password ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задается строкой [16..32] символов.
5	Задать имя класса объектов, среди которых необходимо выполнять поиск пользователей на LDAP-сервере (не обязательно).	<code>RTT(config)# ldap-server search filter user-object-class <NAME></code>	<NAME> – имя класса объектов, задается строкой до 127 символов. Значение по умолчанию: posixAccount.
6	Задать область поиска пользователей в дереве LDAP-сервера (не обязательно).	<code>RTT(config)# ldap-server search scope <SCOPE></code>	<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.

7	Задать интервал, по истечении которого устройство считает, что LDAP-сервер не нашел записей пользователей, подходящих под условие поиска (не обязательно).	RTT(config)# ldap-server search timeout <SEC>	<SEC> – период времени в секундах, принимает значения [0..30] Значение по умолчанию: 0 – устройство ожидает завершения поиска и получения ответа от LDAP-сервера.
8	Задать имя атрибута объекта, со значением которого идет сравнение имени искомого пользователя на LDAP-сервере (не обязательно).	RTT(config)# ldap-server naming-attribute <NAME>	<NAME> – имя атрибута объекта, задаётся строкой до 127 символов. Значение по умолчанию: uid.
9	Задать имя атрибута объекта, значение которого будет определять начальные привилегии пользователя на устройстве (не обязательно).	RTT(config)# ldap-server privilege-level-attribute <NAME>	<NAME> – имя атрибута объекта, задаётся строкой до 127 символов. Значение по умолчанию: priv-lvl
10	Задать глобальное значение кода DSCP для использования в IP-заголовках исходящих пакетов LDAP-сервера (не обязательно).	RTT(config)# ldap-server dscp <DSCP>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63
11	Добавить LDAP-сервер в список используемых серверов и перейти в режим его конфигурирования.	RTT(config)# ldap-server host { <IP-ADDR> <IPV6-ADDR> } [vrf <VRF>] RTT(config-tacacs-server)#	<IP-ADDR> – IP-адрес LDAP-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] <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
12	Указать количество неудачных попыток аутентификации для блокировки логина пользователя и время блокировки (не обязательно)	aaa authentication attempts max-fail <COUNT> <TIME>	<COUNT> – количество неудачных попыток аутентификации, после которых произойдет блокировка пользователя, принимает значения [1..65535]; <TIME> – интервал времени в минутах, на который будет заблокирован пользователь, принимает значения [1..65535]. Значение по умолчанию: <COUNT> - 5; <TIME> - 300

13	Задать номер порта для обмена данными с удаленным LDAP-сервером (не обязательно).	<code>RTT(config-ldap-server) # port <PORT></code>	<PORT> – номер TCP-порта для обмена данными с удаленным сервером, принимает значения [1..65535]. Значение по умолчанию: 389 для LDAP-сервера.
14	Задать приоритет использования удаленного LDAP-сервера (не обязательно).	<code>RTT(config-ldap-server) # priority <PRIORITY></code>	<PRIORITY> – приоритет использования удаленного сервера, принимает значения [1..65535]. Чем ниже значение, тем приоритетнее сервер. Значение по умолчанию: 1.
15	Задать IPv4/IPv6-адрес, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых LDAP-пакетах.	<code>RTT(config-ldap-server) # source-address { <ADDR> <IPV6-ADDR> }</code>	<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, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
16	Указать LDAP в качестве метода аутентификации.	<code>RTT(config) # aaa authentication login { default <NAME> } <METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</code>	<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.
17	Указать LDAP в качестве способа аутентификации повышения привилегий пользователей.	<code>RTT(config) # aaa authentication enable <NAME><METHOD 1> [<METHOD 2>] [<METHOD 3>] [<METHOD 4>]</code>	<NAME> – имя списка строка до 31 символа; default – имя списка по умолчанию. <METHOD> – способы аутентификации: enable – аутентификация с помощью enable-паролей; tacacs – аутентификация по протоколу TACACS; radius – аутентификация по протоколу RADIUS; ldap – аутентификация по протоколу LDAP.

18	Указать способ перебора методов аутентификации в случае отказа.	<code>RTT(config)# aaa authentication mode <MODE></code>	<MODE> – способы перебора методов: chain - если сервер вернул FAIL, переход к следующему в цепочке методу аутентификации; break - если сервер вернул FAIL, прекратить попытки аутентификации. Если сервер недоступен, продолжить попытки аутентификации следующими в цепочке методами. Значение по умолчанию: chain.
19	Перейти в режим конфигурирования соответствующего терминала.	<code>RTT(config)# line <TYPE></code>	<TYPE> – тип консоли: console – локальная консоль; ssh – защищенная удаленная консоль.
20	Активировать список аутентификации входа пользователей в систему.	<code>RTT(config-line-console)# login authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создано на шаге 14.
21	Активировать список аутентификации повышения привилегий пользователей.	<code>RTT(config-line-console)# enable authentication <NAME></code>	<NAME> – имя списка, задаётся строкой до 31 символа. Создано на шаге 15.

7.7.5 Пример настройки аутентификации по telnet через RADIUS-сервер

Задача:

Настроить аутентификацию пользователей, подключающихся по Telnet, через RADIUS (192.168.16.1/24).

Решение:

Настроим подключение к RADIUS-серверу и укажем ключ (password):

```
RTT# configure
RTT(config)# radius-server host 192.168.16.1
RTT(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
RTT(config-radius-server)# exit
```

Создадим профиль аутентификации:

```
RTT(config)# aaa authentication login log radius
```

Укажем режим аутентификации, используемый при подключении по Telnet-протоколу:

```
RTT(config)# line telnet
RTT(config-line-telnet)# login authentication log
RTT(config-line-telnet)# exit
RTT(config)# exit
```

Просмотреть информацию по настройкам подключения к RADIUS-серверу можно командой:

```
RTT# show aaa radius-servers
```

Посмотреть профили аутентификации можно командой:

```
RTT# show aaa authentication
```

7.8 Настройка привилегий команд

Настройка привилегий команд является гибким инструментом, который позволяет назначить набору команд минимально необходимый уровень пользовательских привилегий (1-15). В дальнейшем при создании пользователя можно задать уровень привилегий, определяя ему доступный набор команд.

- 1-9 уровни – позволяют использовать все команды мониторинга (show ...);
- 10-14 уровни – позволяют использовать все команды кроме команд перезагрузки устройства, управления пользователями и ряда других;
- 15 уровень – позволяет использовать все команды.

7.8.1 Алгоритм настройки

Для изменения минимального уровня привилегий необходимого для выполнения команды CLI используется команда:

```
RTT(config)# privilege <COMMAND-MODE> level <PRIV><COMMAND>
```

<COMMAND-MODE> – командный режим;

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

<COMMAND> – поддерево команд, задается строкой до 255 символов.

7.8.2 Пример настройки привилегий команд

Задача:

Перевести все команды просмотра информации об интерфейсах на уровень привилегий 10, кроме команды «show interfaces bridges». Команду «show interfaces bridges» перевести на уровень привилегий 3.

Решение:

В режиме конфигурирования определим команды, разрешенные на использование с уровнем привилегий 10 и уровнем привилегий 3:

```
RTT(config)# privilege root level 3 "show interfaces bridge"
RTT(config)# privilege root level 10 "show interfaces"
```

7.9 Настройка DHCP-сервера

Встроенный DHCP-сервер маршрутизатора может быть использован для настройки сетевых параметров устройств в локальной сети. DHCP-сервер маршрутизаторов способен передавать дополнительные опции на сетевые устройства, например:

- default-router – IP-адрес маршрутизатора, используемого в качестве шлюза по умолчанию;
- domain-name – доменное имя, которое должен будет использовать клиент при разрешении имен хостов через Систему Доменных Имен (DNS);
- dns-server – список адресов серверов доменных имен в данной сети, о которых должен знать клиент. Адреса серверов в списке располагаются в порядке убывания предпочтения.

7.9.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить IPv4/IPv6 DHCP-сервер.	RTT(config)# ip dhcp-server [vrf <VRF>]	<VRF> – имя экземпляра VRF, в рамках которого будет работать DHCP-сервер. Задается строкой до 31 символа.
		RTT(config)# ipv6 dhcp-server [vrf <VRF>]	
2	Задать значение кода DSCP для использования в IP-заголовке исходящих пакетов DHCP-сервера (не обязательно).	RTT(config)# ip dhcp-server dscp <DSCP>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 61.
3	Создать пул IPv4/IPv6-адресов DHCP-сервера и перейти в режим его конфигурирования.	RTT(config)# ip dhcp-server pool <NAME> [vrf <VRF>]	<NAME> – имя пула IPv4/IPv6-адресов DHCP-сервера, задается строка до 31 символа. <VRF> – имя экземпляра VRF, в рамках которого будет работать данный пул IP-адресов DHCP-сервера. Задается строкой до 31 символа
		RTT(config)# ipv6 dhcp-server pool <NAME> [vrf <VRF>]	
4	Задать IPv4/IPv6-адрес и	RTT(config-dhcp-server) #	<ADDR/LEN> – IP-адрес и префикс

	маску для подсети, из которой будет выделен пул IPv4/IPv6-адресов.	network <ADDR/LEN>	подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
		RTT (config-ipv6-dhcp-server) # network <IPv6-ADDR/LEN>	<IPv6-ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].
5	Добавить диапазон IPv4/IPv6-адресов к пулу адресов, конфигурируемого DHCP-сервера.	RTT (config-dhcp-server) # address-range <FROM-ADDR>–<TO-ADDR>	<FROM-ADDR> – начальный IP-адрес диапазона; <TO-ADDR> – конечный IP-адрес диапазона, Адреса задаются в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. Можно указать до 32 диапазонов IP-адресов, список задаётся через запятую.
		RTT (config-ipv6-dhcp-server) # address-range <FROM-ADDR>–<TO-ADDR>	<FROM-ADDR> – начальный IPv6-адрес диапазона; <TO-ADDR> – конечный IP-адрес диапазона; Адреса задаются в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
6	Добавить IPv4/IPv6-адрес для определенного физического адреса к пулу адресов конфигурируемого DHCP-сервера (не обязательно).	RTT (config-dhcp-server) # address <ADDR> {mac-address <MAC> client-identifier <CI>}	<ADDR> – IP-адрес клиента, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <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]. <CI> – идентификатор клиента согласно DHCP Option 61. Может быть задан в одном из следующих видов: NN:NN:NN:NN:NN:NN:NN:NN: – идентификатор клиента в шестнадцатеричной форме и мас-адрес клиента; STRING – текстовая строка длиной от 1 до 64 символов.

		RTT (config-ipv6-dhcp-server) # address <ADDR> mac-address <MAC>	<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде 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]
7	Задать список IPv4-адресов шлюзов по умолчанию, которые DHCP-сервер будет сообщать клиентам, используя DHCP-опцию 3.	RTT (config-dhcp-server) # default-router <ADDR>	<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.
8	Задать DNS-имя сетевого домена. Имя домена передаётся клиентам в составе DHCP-опции 15 (не обязательно).	RTT (config-dhcp-server) # domain-name <NAME> RTT (config-ipv6-dhcp-server) # domain-name <NAME>	<NAME> – DNS-имя домена клиента, задаётся строкой до 255 символов.
9	Задать список IPv4/IPv6-адресов DNS-серверов. Список передаётся клиентам в составе DHCP-опции 6 (не обязательно).	RTT (config-dhcp-server) # dns-server <ADDR> RTT (config-ipv6-dhcp-server) # dns-server <IPV6-ADDR>	<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. <IPV6-ADDR> – IPv6-адрес DNS-сервера, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8 IPv6-адресов, список задаётся через запятую
10	Задать максимальное время аренды IP-адресов (не обязательно). Если DHCP-клиент запрашивает время аренды, превосходящее максимальное значение, то будет установлено время, заданное этой командой.	RTT (config-dhcp-server) # max-lease-time <TIME> RTT (config-ipv6-dhcp-server) # max-lease-time <TIME>	<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
11	Задать время аренды, на которое клиенту будет выдан IP-адрес (не	RTT (config-dhcp-server) # default-lease-time <TIME>	<TIME> – максимальное время аренды IP-адреса, задаётся в формате DD:HH:MM, где:

	обязательно). Данное время будет использоваться если клиент не запрашивал определенное время аренды.	<code>RTT(config-ipv6-dhcp-server)# default-lease-time <TIME></code>	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.
12	Создать идентификатор класса поставщика (DHCP Опция 60) (не обязательно).	<code>RTT(config)# ip dhcp-server vendor-class-id <NAME></code> <code>RTT(config)# ipv6 dhcp-server vendor-class-id <NAME></code>	<NAME> – идентификатор класса поставщика, задаётся строкой до 31 символа.
13	Задать специфическую информацию поставщика (DHCP Опция 43).	<code>RTT(config-dhcp-vendor-id)# vendor-specific-options <HEX></code> <code>RTT(config-ipv6-dhcp-vendor-id)# vendor-specific-options <HEX></code>	<HEX> – специфическая информация поставщика, задаётся в шестнадцатеричном формате до 128 символов.
14	Задать IP-адрес NetBIOS-сервера (DHCP опция 44) (не обязательно).	<code>RTT(config-dhcp-server)# netbios-name-server <ADDR></code>	<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.
15	Задать IP-адрес tftp-сервера (DHCP Option 150) (не обязательно).	<code>RTT(config-dhcp-server)# tftp-server <ADDR></code>	<ADDR> – IP-адрес DNS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].

7.9.2 Пример настройки DHCP-сервера

Задача:

Настроить работу DHCP-сервера в локальной сети, относящейся к зоне безопасности «trusted». Задать пул IP-адресов из подсети 192.168.1.0/24 для раздачи клиентам. Задать время аренды адресов 1 день. Настроить передачу клиентам маршрута по умолчанию, доменного имени и адресов DNS-серверов с помощью DHCP-опций.

Решение:

Создадим зону безопасности «**trusted**» и установим принадлежность используемых сетевых интерфейсов к зонам:

```
RTT# configure
RTT(config)# security zone trusted
RTT(config-zone)# exit
```

Создадим пул адресов с именем «**Simple**» и добавим в данный пул адресов диапазон IP-адресов для выдачи в аренду клиентам сервера. Укажем параметры подсети, к которой принадлежит данный пул, и время аренды для выдаваемых адресов:

```
RTT# configure
```

```
RTT(config)# ip dhcp-server pool Simple
RTT(config-dhcp-server)# network 192.168.1.0/24
RTT(config-dhcp-server)# address-range 192.168.1.100-192.168.1.125
RTT(config-dhcp-server)# default-lease-time 1:00:00
```

Сконфигурируем передачу клиентам дополнительных сетевых параметров:

- маршрут по умолчанию: 192.168.1.1;
- имя домена: rustel.loc;
- список DNS-серверов: DNS1: 172.16.0.1, DNS2: 8.8.8.8.

```
RTT(config-dhcp-server)# domain-name "rustel.loc"
RTT(config-dhcp-server)# default-router 192.168.1.1
RTT(config-dhcp-server)# dns-server 172.16.0.1 8.8.8.8
RTT(config-dhcp-server)# exit
```

Для того чтобы DHCP-сервер мог раздавать IP-адреса из конфигурируемого пула, на маршрутизаторе должен быть создан IP-интерфейс, принадлежащий к той же подсети, что и адреса пула.

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# security-zone trusted
RTT(config-if-gi)# ip address 192.168.1.1/24
RTT(config-if-gi)# exit
```

Для разрешения прохождения сообщений протокола DHCP к серверу необходимо создать соответствующие профили портов, включающие порт источника 68 и порт назначения 67, используемые протоколом DHCP, и создать разрешающее правило в политике безопасности для прохождения пакетов протокола UDP:

```
RTT(config)# object-group service dhcp_server
RTT(config-object-group-service)# port-range 67
RTT(config-object-group-service)# exit
RTT(config)# object-group service dhcp_client
RTT(config-object-group-service)# port-range 68
RTT(config-object-group-service)# exit
RTT(config)# security zone-pair trusted self
RTT(config-zone-pair)# rule 30
RTT(config-zone-rule)# match protocol udp
RTT(config-zone-rule)# match source-address any
RTT(config-zone-rule)# match destination-address any
RTT(config-zone-rule)# match source-port dhcp_client
RTT(config-zone-rule)# match destination-port dhcp_server
RTT(config-zone-rule)# action permit
RTT(config-zone-rule)# enable
RTT(config-zone-rule)# exit
RTT(config-zone-pair)# exit
```

Разрешим работу сервера:

```
RTT(config)# ip dhcp-server
RTT(config)# exit
```

Просмотреть список арендованных адресов можно с помощью команды:

```
RTT# show ip dhcp binding
```

Просмотреть сконфигурированные пулы адресов можно командами:

```
RTT# show ip dhcp server pool
RTT# show ip dhcp server pool Simple
```



Конфигурирование настроек для IPv6 производится по аналогии с IPv4.

7.10 Конфигурирование Destination NAT

Функция Destination NAT (DNAT) состоит в преобразовании IP-адреса назначения у пакетов, проходящих через сетевой шлюз.

DNAT используется для перенаправления трафика, идущего на некоторый «виртуальный» адрес в публичной сети, на «реальный» сервер в локальной сети, находящийся за сетевым шлюзом. Эту функцию можно использовать для организации публичного доступа к серверам, находящимся в частной сети и не имеющим публичного сетевого адреса.

7.10.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки сервиса трансляции адресов получателя.	<code>RTT(config)# nat destination</code>	
2	Создать пул IP-адресов и/или TCP/UDP-портов с определённым именем (не обязательно).	<code>RTT(config-dnat)# pool <NAME></code>	<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа.
3	Установить внутренний IP-адрес, на который будет заменяться IP-адрес получателя.	<code>RTT(config-dnat-pool)# ip address <ADDR></code>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
4	Установить внутренний TCP/UDP порт, на который будет заменяться TCP/UDP порт получателя.	<code>RTT(config-dnat-pool)# ip port <PORT></code>	<PORT> – TCP/UDP порт, принимает значения [1..65535].
5	Создать группу правил с определённым именем.	<code>RTT(config-dnat)# ruleset <NAME></code>	<NAME> – имя группы правил, задаётся строкой до 31 символа.
6	Указать экземпляр VRF, в котором будет работать данная группа правил (не обязательно).	<code>RTT(config-dnat-ruleset)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
7	Задать область применения группы правил. Правила будут применяться только для трафика, идущего из определенной зоны или интерфейса.	<code>RTT(config-dnat-ruleset)# from { zone <NAME> interface <IF> tunnel <TUN> default }</code>	<NAME> – имя зоны изоляции; <IF> – имя интерфейса устройства; <TUN> – имя туннеля устройства; default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.

8	Задать правило с определённым номером. Правила обрабатываются в порядке возрастания.	<code>RTT (config-dnat-ruleset) # rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1..10000].
9	Задать профиль IP-адресов {отправителя получателя}, для которых должно срабатывать правило.	<code>RTT (config-dnat-rule) # match [not]¹ {source destination} -address <OBJ-GROUP-NETWORK-NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа. Значение «any» указывает на любой IP-адрес отправителя.
10	Задать профиль сервисов (tcp/udp-портов) {отправителя получателя}, для которых должно срабатывать правило (не обязательно).	<code>RTT (config-dnat-rule) # match [not]¹ {source destination} -port <PORT-SET-NAME></code>	<PORT-SET-NAME> – имя профиля порта, задаётся строкой до 31 символа. Значение «any» указывает на любой TCP/UDP-порт отправителя.
11	Установить имя или номер IP-протокола, для которого должно срабатывать правило (не обязательно).	<code>RTT (config-dnat-rule) # match [not]¹ {protocol <TYPE> protocol-id <ID> }</code>	<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. Значение «any» указывает на любой тип протокола. <ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
12	Задать тип и код сообщений протокола ICMP, для которых должно срабатывать правило (если в качестве протокола выбран ICMP) (не обязательно).	<code>RTT (config-dnat-rule) # match [not]¹ icmp {<ICMP_TYPE><ICMP_CODE> <TYPE-NAME> }</code>	<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255]. <ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. Значение «any» указывает на любой код сообщения. <TYPE-NAME> – имя типа ICMP сообщения.
13	Задать действие «трансляция адреса и порта получателя» для трафика, удовлетворяющего критериям, заданным командами «match».	<code>RTT (config-dnat-rule) # action destination-nat { off pool <NAME> netmap <ADDR/LEN> }</code>	off – трансляция отключена; pool<NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP портов; netmap <ADDR/LEN> – IP-адрес и маска подсети, используемые при трансляции. Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
14	Активировать конфигурируемое правило.	<code>RTT (config-dnat-rule) # enable</code>	

Каждая команда «match» может содержать ключ «not». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

¹ При использовании команды *not* правило будет срабатывать для значений, которые не входят в указанный профиль

Более подробная информация о командах для настройки межсетевого экрана содержится в «Справочнике команд CLI».

7.10.2 Пример настройки Destination NAT

Задача:

Организовать доступ из публичной сети, относящейся к зоне «UNTRUST», к серверу локальной сети в зоне «TRUST». Адрес сервера в локальной сети - 10.1.1.100. Сервер должен быть доступным извне по адресу 1.2.3.4, доступный порт 80.

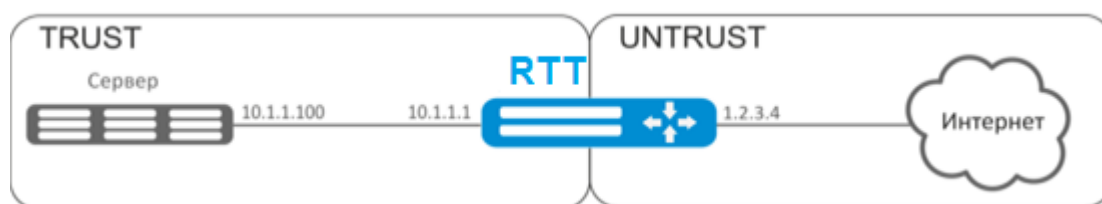


Рисунок 26 – Схема сети

Решение:

Создадим зоны безопасности «UNTRUST» и «TRUST». Установим принадлежность используемых сетевых интерфейсов к зонам. Одновременно назначим IP-адреса интерфейсам.

```
RTT# configure
RTT(config)# security zone UNTRUST
RTT(config-zone)# exit
RTT(config)# security zone TRUST
RTT(config-zone)# exit

RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# security-zone TRUST
RTT(config-if-gi)# ip address 10.1.1.1/25
RTT(config-if-gi)# exit

RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip address 1.2.3.4/29
RTT(config-if-te)# security-zone UNTRUST
RTT(config-if-te)# exit
```

Создадим профили IP-адресов и портов, которые потребуются для настройки правил Firewall и правил DNAT.

- NET_UPLINK – профиль адресов публичной сети;
- SERVER_IP – профиль адресов локальной сети;
- SRV_HTTP – профиль портов.

```
RTT(config)# object-group network NET_UPLINK
RTT(config-object-group-network)# ip address 1.2.3.4
RTT(config-object-group-network)# exit

RTT(config)# object-group service SRV_HTTP
RTT(config-object-group-service)# port 80
RTT(config-object-group-service)# exit
```

```
RTT(config)# object-group network SERVER_IP
RTT(config-object-group-network)# ip address 10.1.1.100
RTT(config-object-group-network)# exit
```

Войдем в режим конфигурирования функции DNAT и создадим пул адресов и портов назначения, в которые будут транслироваться адреса пакетов, поступающие на адрес 1.2.3.4 из внешней сети.

```
RTT(config)# nat destination
RTT(config-dnat)# pool SERVER_POOL
RTT(config-dnat-pool)# ip address 10.1.1.100
RTT(config-dnat-pool)# ip port 80
RTT(config-dnat-pool)# exit
```

Создадим набор правил «DNAT», в соответствии с которыми будет производиться трансляция адресов. В атрибутах набора укажем, что правила применяются только для пакетов, пришедших из зоны «UNTRUST». Набор правил включает в себя требования соответствия данных по адресу и порту назначения (match destination-address, match destination-port) и по протоколу. Кроме этого в наборе задано действие, применяемое к данным, удовлетворяющим всем правилам (action destination-nat). Набор правил вводится в действие командой «enable».

```
RTT(config-dnat)# ruleset DNAT
RTT(config-dnat-ruleset)# from zone UNTRUST
RTT(config-dnat-ruleset)# rule 1
RTT(config-dnat-rule)# match destination-address NET_UPLINK
RTT(config-dnat-rule)# match protocol tcp
RTT(config-dnat-rule)# match destination-port SRV_HTTP
RTT(config-dnat-rule)# action destination-nat pool SERVER_POOL
RTT(config-dnat-rule)# enable
RTT(config-dnat-rule)# exit
RTT(config-dnat-ruleset)# exit
RTT(config-dnat)# exit
```

Для пропуска трафика, идущего из зоны «UNTRUST» в «TRUST», создадим соответствующую пару зон. Пропускать следует только трафик с адресом назначения, соответствующим заданному в профиле «SERVER_IP» и прошедший преобразование DNAT.

```
RTT(config)# security zone-pair UNTRUST TRUST
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address SERVER_IP
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match destination-nat
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# exit
```

Произведенные настройки можно посмотреть с помощью команд:

```
RTT# show ip nat destination pools
RTT# show ip nat destination rulesets
RTT# show ip nat proxy-arp
RTT# show ip nat translations
```

7.11 Конфигурирование Source NAT

Функция Source NAT (SNAT) используется для подмены адреса источника у пакетов, проходящих через сетевой шлюз. При прохождении пакетов из локальной сети в публичную сеть, адрес источника заменяется на один из публичных адресов шлюза. Дополнительно к адресу источника может применяться замена порта источника. При прохождении пакетов из публичной сети в локальную происходит обратная подмена адреса и порта.

Функция SNAT может быть использована для предоставления доступа в Интернет компьютерам, находящимся в локальной сети. При этом не требуется назначения публичных IP-адресов этим компьютерам.

7.11.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим настройки сервиса трансляции адресов отправителя.	<code>RTT(config)# nat source</code>	
2	Создать пул IP-адресов и/или TCP/UDP-портов с определённым именем (не обязательно).	<code>RTT(config-snat)# pool <NAME></code>	<NAME> – имя пула NAT-адресов, задаётся строкой до 31 символа.
3	Установить диапазон IP-адресов, для которых будет заменяться IP-адрес отправителя.	<code>RTT(config-snat-pool)# ip address-range <IP>[-<ENDIP>]</code>	<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.
4	Задать диапазон внешних TCP/UDP портов, на которые будет заменяться TCP/UDP порт отправителя.	<code>RTT(config-snat-pool)# ip port-range <PORT>[-<ENDPORT>]</code>	<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.
5	Установить внутренний TCP/UDP порт, на который будет заменяться TCP/UDP порт отправителя.	<code>RTT(config-snat-pool)# ip port <PORT></code>	<PORT> – TCP/UDP порт, принимает значения [1..65535].
6	Включить функции NAT persistent.	<code>RTT(config-snat-pool)# persistent</code>	
7	Создать группу правил с определённым именем.	<code>RTT(config-snat)# ruleset <NAME></code>	<NAME> – имя группы правил, задаётся строкой до 31 символа.

8	Указать экземпляр VRF, в котором будет работать данная группа правил (не обязательно).	<code>RTT(config-snat-ruleset)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
9	Задать область применения группы правил. Правила будут применяться только для трафика, идущего в определенную зону или интерфейс.	<code>RTT(config-snat-ruleset)# to { zone <NAME> interface <IF> tunnel <TUN> default }</code>	<NAME> – имя зоны изоляции; <IF> – имя интерфейса устройства; <TUN> – имя туннеля устройства default – обозначает группу правил для всего трафика, источник которого не попал под критерии других групп правил.
10	Задать правило с определённым номером. Правила обрабатываются в порядке возрастания.	<code>RTT(config-snat-ruleset)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1..10000].
11	Задать профиль IP-адресов {отправителя получателя}, для которых должно срабатывать правило.	<code>RTT(config-snat-rule)# match [not]¹ {source destination}-address <OBJ-GROUP-NETWORK-NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задается строкой до 31 символа. Значение «any» указывает на любой IP-адрес отправителя.
12	Задать профиль IP-адресов {отправителя получателя}, для которых должно срабатывать правило (не обязательно).	<code>RTT(config-snat-rule)# match [not]¹ {source destination}-port <PORT-SET-NAME></code>	<PORT-SET-NAME> – имя профиля порта, задается строкой до 31 символа. Значение «any» указывает на любой TCP/UDP-порт отправителя.
13	Установить имя или номер IP-протокола, для которого должно срабатывать правило (не обязательно).	<code>RTT(config-snat-rule)# match [not]¹{protocol protocol-id} <TYPE></code>	<TYPE> – тип протокола, принимает значения: esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. Значение «any» указывает на любой тип протокола; <ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
14	Задать тип и код сообщений протокола ICMP, для которых должно срабатывать правило (не обязательно).	<code>RTT(config-snat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> <TYPE-NAME>}</code>	<ICMP_TYPE> – тип сообщения протокола ICMP, принимает значения [0..255]; <ICMP_CODE> – код сообщения протокола ICMP, принимает значения [0..255]. Значение «any» указывает на любой код сообщения; <TYPE-NAME> – имя типа ICMP сообщения

15	Задать действие «трансляция адреса и порта отправителя» для трафика, удовлетворяющего критериям, заданным командами «match»	<pre>RTT(config-snat-rule)# action source-nat { off pool <NAME> netmap <ADDR/LEN> [static] interface [FIRST_PORT – LAST_PORT] }</pre>	<p>off – трансляция отключена; pool<NAME> – имя пула, содержащего набор IP-адресов и/или TCP/UDP портов; netmap <ADDR/LEN> – IP-адрес и маска подсети, используемые при трансляции; static – опция для организации статического NAT.</p> <p>Параметр задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].</p> <p>interface [FIRST_PORT – LAST_PORT] – задаёт трансляцию в IP-адрес интерфейса. Если дополнительно задан диапазон TCP/UDP-портов, то трансляция будет происходить только для TCP/UDP- портов отправителя, входящих в указанный диапазон.</p>
16	Активировать конфигурируемое правило.	<pre>RTT(config-snat-rule)# enable</pre>	

Каждая команда «match» может содержать ключ «not». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

Более подробная информация о командах для настройки межсетевого экрана содержится в «Справочнике команд CLI».

7.11.2 Пример настройки 1

Задача:

Настроить доступ пользователей локальной сети 10.1.2.0/24 к публичной сети с использованием функции Source NAT. Задать диапазон адресов публичной сети для использования SNAT 100.0.0.100-100.0.0.249.

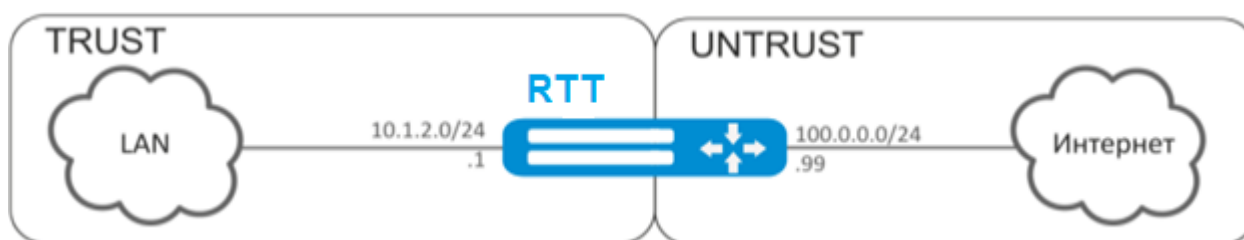


Рисунок 27 – Схема сети

Решение:

Конфигурирование начнем с создания зон безопасности, настройки сетевых интерфейсов и определения их принадлежности к зонам безопасности. Создадим доверенную зону «TRUST» для локальной сети и зону «UNTRUST» для публичной сети.

RTT# **configure**

```
RTT(config)# security zone UNTRUST
RTT(config-zone)# exit
RTT(config)# security zone TRUST
RTT(config-zone)# exit

RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip address 10.1.2.1/24
RTT(config-if-gi)# security-zone TRUST
RTT(config-if-gi)# exit

RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip address 100.0.0.99/24
RTT(config-if-te)# security-zone UNTRUST
RTT(config-if-te)# exit
```

Для конфигурирования функции SNAT и настройки правил зон безопасности потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий адреса, которым разрешен выход в публичную сеть, и профиль адресов публичной сети «PUBLIC_POOL».

```
RTT(config)# object-group network LOCAL_NET
RTT(config-object-group-network)# ip address-range 10.1.2.2-10.1.2.254
RTT(config-object-group-network)# exit

RTT(config)# object-group network PUBLIC_POOL
RTT(config-object-group-network)# ip address-range 100.0.0.100-100.0.0.249
RTT(config-object-group-network)# exit
```

Для пропуска трафика из зоны «TRUST» в зону «UNTRUST» создадим пару зон и добавим правила, разрешающие проходить трафику в этом направлении. Дополнительно включена проверка адреса источника данных на принадлежность к диапазону адресов «LOCAL_NET» для соблюдения ограничения на выход в публичную сеть. Действие правил разрешается командой **enable**.

```
RTT(config)# security zone-pair TRUST UNTRUST
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# match source-address LOCAL_NET
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

Конфигурируем сервис SNAT. Первым шагом создаётся пул адресов публичной сети, используемых для сервиса SNAT.

```
RTT(config)# nat source
RTT(config-snat)# pool TRANSLATE_ADDRESS
RTT(config-snat-pool)# ip address-range 100.0.0.100-100.0.0.249
RTT(config-snat-pool)# exit
```

Вторым шагом создаётся набор правил SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть – в зону «UNTRUST». Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET».

```
RTT(config-snat)# ruleset SNAT
RTT(config-snat-ruleset)# to zone UNTRUST
RTT(config-snat-ruleset)# rule 1
RTT(config-snat-rule)# match source-address LOCAL_NET
RTT(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
RTT(config-snat-rule)# enable
```

```
RTT(config-snat-rule)# exit
RTT(config-snat-ruleset)# exit
```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в публичный пул, необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов публичной сети «PUBLIC_POOL».

```
RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

Для того чтобы устройства локальной сети могли получить доступ к публичной сети, на них должна быть настроена маршрутизация – адрес 10.1.2.1 должен быть назначен адресом шлюза.

На самом маршрутизаторе также должен быть создан маршрут для направления на публичную сеть. Этот маршрут может быть назначен маршрутом по умолчанию с помощью следующей команды.

```
RTT(config)# ip route 0.0.0.0/0 100.0.0.1
RTT(config)# exit
```

7.11.3 Пример настройки 2

Задача:

Настроить доступ пользователей локальной сети 21.12.2.0/24 к публичной сети с использованием функции Source NAT без использования межсетевого экрана (firewall). Диапазон адресов публичной сети для использования SNAT 200.10.0.100-200.10.0.249.



Рисунок 28 – Схема сети

Решение:

Конфигурирование начнем с настройки сетевых интерфейсов и отключения межсетевого экрана:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip address 21.12.2.1/24
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# exit

RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip address 200.10.0.1/24
RTT(config-if-te)# ip firewall disable
RTT(config-if-te)# exit
```

Для конфигурирования функции SNAT потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий адреса, которым разрешен выход в публичную сеть, и профиль адресов публичной сети «PUBLIC_POOL»:

```
RTT(config)# object-group network LOCAL_NET
RTT(config-object-group-network)# ip address-range 21.12.2.2-21.12.2.254
RTT(config-object-group-network)# exit

RTT(config)# object-group network PUBLIC_POOL
RTT(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.249
RTT(config-object-group-network)# exit
```

Конфигурируем сервис SNAT.

Первым шагом создаётся пул адресов публичной сети, используемых для сервиса SNAT:

```
RTT(config)# nat source
RTT(config-snat)# pool TRANSLATE_ADDRESS
RTT(config-snat-pool)# ip address-range 200.10.0.100-200.10.0.249
RTT(config-snat-pool)# exit
```

Вторым шагом создаётся набор правил SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть через порт te1/0/1. Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET»:

```
RTT(config-snat)# ruleset SNAT
RTT(config-snat-ruleset)# to interface te1/0/1
RTT(config-snat-ruleset)# rule 1
RTT(config-snat-rule)# match source-address LOCAL_NET
RTT(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
RTT(config-snat-rule)# enable
RTT(config-snat-rule)# exit
RTT(config-snat-ruleset)# exit
```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в публичный пул, необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов публичной сети «PUBLIC_POOL»:

```
RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

Для того чтобы устройства локальной сети могли получить доступ к публичной сети, на них должна быть настроена маршрутизация – адрес 21.12.2.1 должен быть назначен адресом шлюза.

На самом маршрутизаторе также должен быть создан маршрут для направления на публичную сеть. Этот маршрут может быть назначен маршрутом по умолчанию с помощью следующей команды:

```
RTT(config)# ip route 0.0.0.0/0 200.10.0.254
RTT(config)# exit
```

7.12 Конфигурирование Static NAT

Static NAT — статический NAT задает однозначное соответствие одного адреса другому. Иными словами, при прохождении через маршрутизатор, адрес меняется на другой строго заданный адрес, один-к-одному. Запись о такой трансляции хранится неограниченно долго, пока не будет произведена перенастройка NAT на маршрутизаторе.

7.10.1 Алгоритм настройки

Настройка Static NAT осуществляется средствами Source NAT, алгоритм настройки которой описан в разделе 7.11.1 настоящего руководства.

7.10.2 Пример настройки Static NAT

Задача:

Настроить двухстороннюю и постоянную трансляцию из локальной сети для диапазона адресов 21.12.2.100-21.12.2.150 в публичную сеть 200.10.0.0/24. Диапазон адресов публичной сети для использования трансляции – 200.10.0.100-200.10.0.150.



Рисунок 29 – Схема сети

Решение:

Начнем конфигурирование с настройки сетевых интерфейсов и отключения межсетевого экрана:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip address 21.12.2.1/24
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# exit

RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip address 200.10.0.1/24
RTT(config-if-te)# ip firewall disable
RTT(config-if-te)# exit
```

Для конфигурирования Static NAT потребуется создать профиль адресов локальной сети «LOCAL_NET», включающий локальную подсеть, и профиль адресов публичной сети «PUBLIC_POOL»:

```
RTT(config)# object-group network LOCAL_NET
RTT(config-object-group-network)# ip prefix 21.12.2.0/24
RTT(config-object-group-network)# exit

RTT(config)# object-group network PUBLIC_POOL
RTT(config-object-group-network)# ip prefix 200.10.0.0/24
RTT(config-object-group-network)# exit
```

Диапазон адресов публичной сети для использования Static NAT задаем в профиле «PROXY»:

```
RTT(config)# object-group network PROXY
RTT(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.150
RTT(config-object-group-network)# exit
```

Конфигурируем сервис Static NAT в режиме конфигурирования SNAT. В атрибутах набора укажем, что правила применяются только для пакетов, направляющихся в публичную сеть через порт te1/0/1. Правила включают проверку адреса источника данных на принадлежность к пулу «LOCAL_NET» и проверку адресов назначения на принадлежность к пулу «PUBLIC_POOL».

```
RTT(config)# nat source
RTT(config-snat)# ruleset SNAT
RTT(config-snat-ruleset)# to interface te1/0/1
RTT(config-snat-ruleset)# rule 1
RTT(config-snat-rule)# match source-address LOCAL_NET
RTT(config-snat-rule)# match destination-address PUBLIC_POOL
RTT(config-snat-rule)# action source-nat netmap 200.10.0.0/24 static
RTT(config-snat-rule)# enable
RTT(config-snat-rule)# exit
RTT(config-snat-ruleset)# exit
```

Для того чтобы маршрутизатор отвечал на запросы протокола ARP для адресов, входящих в пул трансляции «PROXY», необходимо запустить сервис ARP Proxy. Сервис ARP Proxy настраивается на интерфейсе, которому принадлежит IP-адрес из подсети профиля адресов «PROXY».

```
RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if-te)# ip nat proxy-arp PROXY
```

Для того чтобы устройства локальной сети могли получить доступ к сети 200.10.0.0/24, на них должна быть настроена маршрутизация – адрес 21.12.2.1 должен быть назначен адресом шлюза.

Изменения конфигурации вступают в действие по команде применения.

```
RTT# commit
Configuration has been successfully committed
RTT# confirm
Configuration has been successfully confirmed
```

Посмотреть активные трансляции можно с помощью команды:

```
RTT# show ip nat translations
```

7.10.3 Пример настройки фильтрации приложений (DPI)



Внимание! Использование механизма фильтрации приложений многократно снижает производительность маршрутизатора из-за необходимости проверки каждого пакета. Производительность снижается с ростом количества выбранных приложений для фильтрации.

Задача:

Блокировать доступ к ресурсам youtube, bittorrent и facebook.



Рисунок 30 – Схема сети

Решение:

Для каждой сети RTT создадим свою зону безопасности:

```
RTT# configure
RTT(config)# security zone LAN
RTT(config-zone)# exit
RTT(config)# security zone WAN
RTT(config-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
RTT(config)# interface gi1/0/1
RTT(config-if-gi)# ip address 10.0.0.1/24
RTT(config-if-gi)# security-zone WAN
RTT(config-if-gi)# exit
RTT(config)# interface gi1/0/2
RTT(config-if-te)# ip address 192.168.0.1/24
RTT(config-if-te)# security-zone LAN
RTT(config-if-te)# exit
```

Для настройки правил зон безопасности потребуется создать профиль приложений, которые необходимо будет блокировать.

```
RTT(config)# object-group application APP
RTT(config-object-group-application)# application youtube
RTT(config-object-group-application)# application bittorrent
RTT(config-object-group-application)# application facebook
RTT(config-object-group-application)# exit
```

Для установки правил прохождения трафика из зоны «WAN» в зону «LAN» создадим пару зон и добавим правило, запрещающее проходить трафику приложений, и правило, разрешающее проходить остальному трафику. Действие правил разрешается командой *enable*:

```
RTT(config)# security zone-pair WAN LAN
```

```

RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action deny
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match application APP
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# rule 2
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair-pair)# exit

```

Для установки правил прохождения трафика из зоны «LAN» в зону «WAN» создадим пару зон и добавим правило, разрешающее прохождение всего трафика. Действие правил разрешается командой *enable*:

```

RTT(config)# security zone-pair LAN WAN
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair-pair)# exit

```

Посмотреть членство портов в зонах можно с помощью команды:

```
RTT# show security zone
```

Посмотреть пары зон и их конфигурацию можно с помощью команд:

```

RTT# show security zone-pair
RTT# show security zone-pair configuration

```

Посмотреть активные сессии можно с помощью команд:

```
RTT# show ip firewall sessions
```

7.13 Настройка логирования и защиты от сетевых атак

7.13.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить защиту от ICMP flood атак.	RTT(config)# ip firewall screen dos- defense icmp- threshold { <NUM> }	<NUM> – количество ICMP-пакетов в секунду задается в диапазоне [1..10000]

2	Включить защиту от land атак	<code>RTT(config)# firewall screen dos-defense land</code>	
3	Включить ограничение количества одновременных сессий на основании адреса назначения	<code>RTT(config)# ip firewall screen dos-defense limit-session-destination { <NUM> }</code>	<NUM> – ограничение количества IP-сессий задается в диапазоне [1..10000].
4	Включить ограничение количества одновременных сессий на основании адреса источника, которое смягчает DoS атаки	<code>RTT(config)# ip firewall screen dos-defense limit-session-source { <NUM> }</code>	<NUM> – ограничение количества IP -сессий задается в диапазоне [1..10000].
5	Включить защиту от SYN flood атак.	<code>RTT(config)# ip firewall screen dos-defense syn-flood { <NUM> } [src-dst]</code>	<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.
6	Включить защиту от UDP flood атак.	<code>RTT(config)# ip firewall screen dos-defense udp-threshold { <NUM> }</code>	<NUM> – максимальное количество UDP пакетов в секунду задается в диапазоне [1..10000].
7	Включить защиту от winnuke атак	<code>RTT(config)# ip firewall screen dos-defense winnuke</code>	
8	Включить блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK.	<code>RTT(config)# ip firewall screen spy-blocking fin-no-ack</code>	
9	Включить блокировку ICMP-пакетов различных типов	<code>RTT(config)# ip firewall screen spy-blocking icmp-type</code>	<TYPE> – тип ICMP, может принимать значения: destination-unreachable echo-request reserved source-quench time-exceeded
10	Включить защиту от IP-sweep атак	<code>RTT(config)# ip firewall screen spy-blocking ip-sweep { <NUM> }</code>	<NUM> – интервал выявления ip sweep атаки, задается в миллисекундах [1..1000000].
11	Включить защиту от port scan атак	<code>RTT(config)# ip firewall screen spy-blocking port-scan { <threshold> } [<TIME>]</code>	<threshold> – интервал в миллисекундах, в течении которого будет фиксироваться port scan атака [1..1000000]. <TIME> – время блокировки в миллисекундах [1..1000000].

12	Включить защиту от IP spoofing атак	RTT(config)# ip firewall screen spy- blocking spoofing	
13	Включить блокировку TCP-пакетов, с установленными флагами SYN и FIN.	RTT(config)# ip firewall screen spy- blocking syn-fin	
14	Включить блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN,PSH,URG. Данной командой обеспечивается защита от атаки XMAS	RTT(config)# ip firewall screen spy- blocking tcp-all-flag	
15	Включить блокировку TCP-пакетов, с нулевым полем flags.	RTT(config)# ip firewall screen spy- blocking tcp-no-flag	
16	Включить блокировку фрагментированных ICMP-пакетов	RTT(config)# ip firewall screen suspicious-packets icmp-fragment	
17	Включить блокировку фрагментированных IP пакетов.	RTT(config)# ip firewall screen suspicious-packets ip-fragment	
18	Включить блокировку ICMP-пакетов длиной более 1024 байт.	RTT(config)# ip firewall screen suspicious-packets icmp-fragment	
19	Включить блокировку фрагментированных TCP-пакетов, с флагом SYN	RTT(config)# ip firewall screen suspicious-packets syn-fragment	
20	Включить блокировку фрагментированных UDP-пакетов	RTT(config)# ip firewall screen suspicious-packets udp-fragment	
21	Включить блокировку пакетов, с ID протокола в заголовке IP равном 137 и более	RTT(config)# ip firewall screen suspicious-packets unknown-protocols	
22	Установить частоту оповещения (по SNMP, syslog и в CLI) об обнаруженных и отраженных сетевых атаках	RTT(config)# ip firewall logging interval <NUM>	<NUM> – интервал времени в секундах [30 .. 2147483647]
23	Включить более детальный вывод сообщений по обнаруженным и отраженным сетевым атакам в CLI.	RTT(config)# ip firewall logging screen detailed	

24	Включить механизм обнаружения и логирования DoS атак через CLI, syslog и по SNMP.	RTT(config)# ip firewall logging screen dos-defense <ATTACK_TYPE>	<ATTACK_TYPE> – тип DoS атаки, принимает значения: icmp-threshold, land, limit-session-destination, limit-session-source, syn-flood, udp-threshold, winnuke.
25	Включить механизм обнаружения и логирования шпионской активности через CLI, syslog и по SNMP	RTT(config)# ip firewall logging screen spy-blocking { <ATTACK_TYPE> icmp-type <ICMP_TYPE> }	<ATTACK_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.
26	Включить механизм обнаружения нестандартных пакетов и логирования через CLI, syslog и по SNMP	RTT(config)# ip firewall logging screen suspicious- packets <PACKET_TYPE>	<PACKET_TYPE> – тип нестандартных пакетов, принимает значения: icmp-fragment, ip-fragment, large-icmp, syn-fragment, udp-fragment, unknown-protocols.

7.13.2 Описание механизмов защиты от 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 адреса назначения. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый запрос и отвечать на него.

firewall screen dos-defense land

Данная команда включает защиту от land атак. При включенной защите блокируются пакеты с одинаковыми source и destination IP-адресами, и флагом SYN в заголовке TCP. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый TCP SYN пакет и попыток хоста установить TCP сессию с самим собой.

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 атаках: SYN flood, UDP flood, ICMP flood, и т.д.). Команда включает ограничение количества одновременных сессий на основании адреса назначения, которое смягчает DoS атаки.

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 количества одновременных сессий на основании адреса источника, которое смягчает DoS атаки.

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. Атака приводит к перегрузке хоста и выводу его из строя из-за необходимости обрабатывать каждый TCP SYN пакет и попыток установить TCP-сессии.

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. Атака приводит к перегрузке хоста и выводу его из строя из-за массивного UDP-трафика.

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. Атака приводит к выходу из строя старых версий Windows (до 95 версии).

ip firewall screen spy-blocking fin-no-ack

Данная команда включает блокировку TCP-пакетов с установленным флагом FIN и не установленным флагом ACK. Такие пакеты являются нестандартными и по ответу можно определить операционную систему жертвы.

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов

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. Защита не дает злоумышленнику узнать о топологии сети и доступности хостов.

ip firewall screen spy-blocking port-scan

Данная команда включает защиту от port scan атак. Если в течение первого заданного интервала времени (<THRESHOLD>) на один источник приходит более 10 TCP-пакетов с флагом SYN на разные TCP-порты, то такое поведение фиксируется как port scan атака и все последующие пакеты такого рода от источника блокируются на второй заданный интервал времени (<TIME>). Злоумышленник не сможет быстро просканировать открытые порты на устройстве.

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. Защищает от вторжений в сеть с подмененными source IP-адресами.

ip firewall screen spy-blocking syn-fin

Данная команда включает блокировку TCP-пакетов, с установленными флагами SYN и FIN. Такие пакеты являются нестандартными и по ответу можно определить операционную систему жертвы.

ip firewall screen spy-blocking tcp-all-flag

Данная команда включает блокировку TCP-пакетов, со всеми флагами или с набором флагов: FIN,PSH,URG. Обеспечивается защита от атаки XMAS.

ip firewall screen spy-blocking tcp-no-flag

Данная команда включает блокировку TCP-пакетов, с нулевым полем flags. Такие пакеты являются нестандартными и по ответу можно определить операционную систему жертвы.

ip firewall screen suspicious-packets icmp-fragment

Данная команда включает блокировку фрагментированных ICMP-пакетов. ICMP-пакеты обычно небольшого размера и необходимости в их фрагментировании нет.

ip firewall screen suspicious-packets ip-fragment

Данная команда включает блокировку фрагментированных пакетов.

ip firewall screen suspicious-packets large-icmp

Данная команда включает блокировку ICMP-пакетов длиной более 1024 байт.

ip firewall screen suspicious-packets syn-fragment

Данная команда включает блокировку фрагментированных TCP-пакетов, с флагом SYN. TCP пакеты с SYN флагом обычно небольшого размера и необходимости в их фрагментировании нет. Защита предотвращает накопление фрагментированных пакетов в буфере.

ip firewall screen suspicious-packets udp-fragment

Данная команда включает блокировку фрагментированных UDP-пакетов.

ip firewall screen suspicious-packets unknown-protocols

Данная команда включает блокировку пакетов, с ID протокола в заголовке IP равном 137 и более.

7.13.3 Пример настройки логирования и защиты от сетевых атак

Задача:

Необходимо защитить LAN сеть и маршрутизатор RTT от сетевых атак land, syn-flood, ICMP flood и настроить оповещение об атаках по SNMP на SNMP-сервер 192.168.0.10



Рисунок 31 – Схема сети

Решение:

Предварительно необходимо настроить интерфейсы и firewall (настройка firewall или ее отсутствие не повлияют на работу защиты от сетевых атак):

```
RTT(config)# security zone LAN
RTT(config-zone)# exit
RTT(config)# security zone WAN
RTT(config-zone)# exit
RTT(config)# security zone-pair LAN WAN
RTT(config-zone-pair)# rule 100
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# ex
RTT(config-zone-pair)# exit
RTT(config)# security zone-pair WAN LAN
RTT(config-zone-pair)# rule 100
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# ip address 192.168.0.1/24
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# security-zone WAN
RTT(config-if-gi)# ip address 10.0.0.1/24
RTT(config-if-gi)# exit
```

Настроим защиту от land, syn-flood, ICMP flood: атак

```
RTT(config)# ip firewall screen dos-defense land
RTT(config)# ip firewall screen dos-defense syn-flood 100 src-dst
RTT(config)# ip firewall screen dos-defense icmp-threshold 100
```

Настроим логирование обнаруженных атак:

```
RTT(config)# ip firewall logging screen dos-defense land
RTT(config)# ip firewall logging screen dos-defense syn-flood
RTT(config)# ip firewall logging screen dos-defense icmp-threshold
```

Настроим SNMP-сервер, на который будут отправляться трапы

```
RTT(config)# snmp-server
RTT(config)# snmp-server host 192.168.0.10
```

Посмотреть статистику по зафиксированным сетевым атакам можно командой:

```
RTT# show ip firewall screen counters
```

7.14 Конфигурирование Firewall

Firewall – комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами.

7.14.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать зоны безопасности.	RTT(config)# security zone <zone-name1> RTT(config)# security zone <zone-name2>	<zone-name> - до 12 символов.
2	Задать описание зоны безопасности.	RTT(config-zone)# description <description>	<description> - до 255 символов.
3	Указать экземпляр VRF, в котором будет работать данная зона безопасности (не обязательно).	RTT(config-zone)# ip vrf forwarding <VRF>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Включить счетчики сессий для NAT и Firewall (не обязательно, снижает производительность).	RTT(config)# ip firewall sessions counters	
5	Отключить фильтрацию пакетов, для которых не удалось определить принадлежность к какому-либо известному соединению и которые не являются началом нового соединения (не обязательно, снижает производительность).	RTT(config)# ip firewall sessions allow-unknown	
6	Выбрать режима работы межсетевого экрана (не обязательно)	RTT(config)# ip firewall mode <MODE>	<MODE> – режим работы межсетевого экрана, может принимать значения: stateful, stateless. Значение по умолчанию: stateful
7	Определить время жизни сессии для неподдерживаемых протоколов (не обязательно).	RTT(config)# ip firewall sessions generic-timeout <TIME>	<TIME> – время жизни сессии для неподдерживаемых протоколов, принимает значения в секундах [1..8553600]. По умолчанию: 60 секунд.
8	Определить время жизни ICMP-сессии, по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions icmp-timeout <TIME>	<TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
9	Определить время жизни ICMPv6-сессии, по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions icmpv6-timeout <TIME>	<TIME> – время жизни ICMP-сессии, принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
10	Определить размер таблицы сессий ожидающих обработки (не обязательно).	RTT(config)# ip firewall sessions max-expect <COUNT>	<COUNT> – размер таблицы, принимает значения [1..8553600]. По умолчанию: 256.

11	Определить размер таблицы отслеживаемых сессий (не обязательно).	RTT(config)# ip firewall sessions max-tracking <COUNT>	<COUNT> – размер таблицы, принимает значения [1..8553600]. По умолчанию: 512000.
12	Определить время жизни TCP-сессии в состоянии «соединение устанавливается», по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions tcp-connect-timeout <TIME>	<TIME> – время жизни TCP-сессии в состоянии "соединение устанавливается", принимает значения в секундах [1..8553600]. По умолчанию: 60 секунд.
13	Определить время жизни TCP-сессии в состоянии "соединение закрывается", по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions tcp-disconnect- timeout <TIME>	<TIME> – время жизни TCP-сессии в состоянии "соединение закрывается", принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
14	Определить время жизни TCP-сессии в состоянии "соединение установлено", по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions tcp-established- timeout <TIME>	<TIME> – время жизни TCP-сессии в состоянии "соединение установлено", принимает значения в секундах [1..8553600]. По умолчанию: 120 секунд.
15	Определить время ожидания, по истечении которого происходит фактическое удаление закрытой TCP-сессии из таблицы отслеживаемых сессий (не обязательно).	RTT(config)# ip firewall sessions tcp-latecome-timeout <TIME>	<TIME> – время ожидания, принимает значения в секундах [1..8553600]. По умолчанию: 120 секунд.
16	Включить функцию отслеживания сессий уровня приложений для отдельных протоколов (не обязательно).	RTT(config)# ip firewall sessions tracking { <PROTOCOL> sip [<OBJECT-GROUP- SERVICE>] }	<PROTOCOL> - протокол уровня приложений [ftp, h323, pptp, netbios-ns, tftp] сессии которого должны отслеживаться. <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. По умолчанию - отключено для всех протоколов

17	Определить время жизни UDP-сессии в состоянии "соединение подтверждено", по истечении которого она считается устаревшей (не обязательно).	RTT(config)# ip firewall sessions udp-assured-timeout <TIME>	<TIME> – время жизни UDP-сессии в состоянии "соединение подтверждено", принимает значения в секундах [1..8553600]. По умолчанию: 180 секунд.
18	Определить время жизни UDP-сессии в состоянии «соединение не подтверждено», по истечении которого она считается устаревшей.	RTT(config)# ip firewall sessions udp-wait-timeout <TIME>	<TIME> – время жизни UDP-сессии в состоянии «соединение не подтверждено», принимает значения в секундах [1..8553600]. По умолчанию: 30 секунд.
19	Создать списки IP-адресов, которые будут использоваться при фильтрации.	RTT(config)# object- group network <obj- group-name>	<obj-group-name> – до 31 символа.
20	Задать описание списка IP-адресов (не обязательно).	RTT(config-object- group-network)# description <description>	<description> – описание профиля, задается строкой до 255 символов.
21	Внести необходимые IPv4/IPv6- адреса в список.	RTT(config-object- group-network)# ip prefix <ADDR/LEN>	<ADDR/LEN> – подсеть, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
		RTT(config-object- group-network)# ip address-range <FROM- ADDR>–<TO-ADDR>	<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].
		RTT(config-object- group-network)# ipv6 prefix <IPV6- ADDR/LEN>	<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].
		RTT(config-object- group-network)# ipv6 address-range <FROM- ADDR>–<TO-ADDR>	<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, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

22	Создать списки сервисов, которые будут использоваться при фильтрации.	<code>RTT(config)# object-group service <obj-group-name></code>	<obj-group-name> – имя профиля сервисов, задается строкой до 31 символа.
23	Задать описание списка сервисов (не обязательно).	<code>RTT(config-object-group-service)# description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
24	Внести необходимые сервисы (tcp/udp порты) в список.	<code>RTT(config-object-group-service)# port-range <port></code>	<port> – принимает значение [1..65535]. Можно указать несколько портов перечислением через запятую «,» либо указать диапазон портов через «-».
25	Создать списки приложений, которые будут использоваться в механизме DPI.	<code>RTT(config)# object-group application <NAME></code>	<NAME> – имя профиля приложений, задается строкой до 31 символа.
26	Задать описание списка приложений (не обязательно).	<code>RTT(config-object-group-application)# description <description></code>	<description> – описание профиля, задается строкой до 255 символов.
27	Внести необходимые приложения в списки.	<code>RTT(config-object-group-application)# application <APPLICATION ></code>	< APPLICATION > – указывает приложение подпадающее под действие данного профиля
28	Включить интерфейсы (физические, логические, E1/Multilink и подключаемые), сервер удаленного доступа (l2tp, openvpn, pptp) или туннели (gre, ip4ip4, l2tp, lt, pppe, pptp) в зоны безопасности (если необходимо).	<code>RTT(config-if-gi)# security-zone <zone-name></code>	<zone-name> – до 12 символов.
	Отключить функции Firewall на сетевом интерфейсе (физические, логические, E1/Multilink и подключаемые), сервере удаленного доступа (l2tp, openvpn, pptp) или туннели (gre, ip4ip4, l2tp, lt, pppe, pptp) (если необходимо)	<code>RTT(config-if-gi)# ip firewall disable</code>	
29	Создать набор правил межзонового взаимодействия.	<code>RTT(config)# security zone-pair <src-zone-name1> <dst-zone-name2></code>	<src-zone-name> - до 12 символов. <dst-zone-name> - до 12 символов.
30	Создать правило межзонового взаимодействия.	<code>RTT(config-zone-pair)# rule <rule-number></code>	<rule-number> - 1..10000.
31	Задать описание правила (не обязательно).	<code>RTT(config-zone-rule)# description <description></code>	<description> - до 255 символов.

32	Указать действие данного правила.	RTT (config-zone-rule)# action <action> [log]	<action> - permit/deny/reject/netflow-sample/sflow-sample log – ключ для активации логирования сессий, устанавливаемым согласно данному правилу.
33	Установить имя или номер IP-протокола, для которого должно срабатывать правило.	RTT (config-zone-rule)# match [not] ¹ protocol <protocol-type>	<protocol-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.
		RTT (config-zone-rule)# match [not] ¹ protocol-id <protocol-id>	<protocol-id> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
34	Установить профиль IP-адресов отправителя, для которых должно срабатывать правило.	RTT (config-zone-rule)# match [not] ¹ source-address <OBJ-GROUP-NETWORK-NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя/получателя.
35	Установить профиль IP-адресов получателя, для которых должно срабатывать правило.	RTT (config-zone-rule)# match [not] ¹ destination-address <OBJ-GROUP-NETWORK-NAME>	
36	Установить MAC-адрес отправителя, для которого должно срабатывать правило (не обязательно).	RTT (config-zone-rule)# match [not] ¹ source-mac <mac-addr>	<mac-addr> – задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].
37	Установить MAC-адрес получателя, для которого должно срабатывать правило (не обязательно).	RTT (config-zone-rule)# match [not] ¹ destination-mac <mac-addr>	
38	Установить профиль TCP/UDP-портов отправителя, для которых должно срабатывать правило (если указан протокол).	RTT (config-zone-rule)# match [not] ¹ source-port <PORT-SET-NAME>	<PORT-SET-NAME> – задаётся строкой до 31 символа. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя/получателя.
39	Установить профиль TCP/UDP-портов получателя, для которых должно срабатывать правило (если указан протокол).	RTT (config-zone-rule)# match [not] ¹ destination-port <PORT-SET-NAME>	
40	Установить тип и код сообщений протокола ICMP, для которых должно срабатывать правило (если в качестве протокола выбран ICMP).	RTT (config-zone-rule)# match [not] ¹ icmp <ICMP_TYPE> <ICMP_CODE>	<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.

41	Установить ограничение, при котором правило будет срабатывать только для трафика, измененного сервисом трансляции IP-адресов и портов получателя.	<code>RTT(config-zone-rule)# match [not]¹ destination-nat</code>	
42	Установить максимальную скорость прохождения пакетов (не обязательно, доступно только для zone-pair any self и zone-pair <zone-name> any).	<code>RTT(config-zone-pair-rule)# rate-limit pps <rate-pps></code>	<rate-pps> - максимальное количество пакетов, которое может быть передано. Принимает значения [1..10000].
43	Установить фильтрацию только для фрагментированных IP-пакетов (не обязательно, доступно только для zone-pair any self и zone-pair <zone-name> any).	<code>RTT(config-zone-pair-rule)# match [not]¹ fragment</code>	
44	Установить фильтрацию для IP-пакетов, содержащих ip-option (не обязательно, доступно только для zone-pair any self и zone-pair <zone-name> any).	<code>RTT(config-zone-pair-rule)# match [not]¹ ip-option</code>	
45	Включить правило межзонового взаимодействия.	<code>RTT(config-zone-rule)# enable</code>	
46	Активировать фильтрацию и режим отслеживания сессий при прохождении пакетов между участниками одной Bridge-группы (не обязательно, доступно только на RTT-R800)	<code>RTT(config-bridge)# ports firewall enable</code>	

Каждая команда «match» может содержать ключ «not». При использовании данного ключа под правило будут подпадать пакеты, не удовлетворяющие заданному критерию.

Более подробная информация о командах для настройки межсетевого экрана содержится в «Справочнике команд CLI».

7.14.2 Пример настройки Firewall

Задача:

Разрешить обмен сообщениями по протоколу ICMP между устройствами ПК1, ПК2 и маршрутизатором RTT.

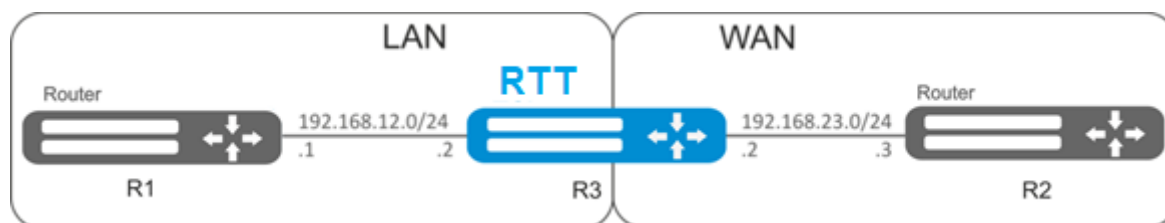


Рисунок 32 – Схема сети

Решение:

Для каждой сети RTT создадим свою зону безопасности:

```
RTT# configure
RTT(config)# security zone LAN
RTT(config-zone)# exit
RTT(config)# security zone WAN
RTT(config-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
RTT(config)# interface gi1/0/2
RTT(config-if-gi)# ip address 192.168.12.2/24
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# exit
RTT(config)# interface gi1/0/3
RTT(config-if-gi)# ip address 192.168.23.2/24
RTT(config-if-gi)# security-zone WAN
RTT(config-if-gi)# exit
```

Для настройки правил зон безопасности потребуется создать профиль адресов сети «LAN», включающий адреса, которым разрешен выход в сеть «WAN», и профиль адресов сети «WAN».

```
RTT(config)# object-group network WAN
RTT(config-object-group-network)# ip address-range 192.168.23.2
RTT(config-object-group-network)# exit
RTT(config)# object-group network LAN
RTT(config-object-group-network)# ip address-range 192.168.12.2
RTT(config-object-group-network)# exit
RTT(config)# object-group network LAN_GATEWAY
RTT(config-object-group-network)# ip address-range 192.168.12.1
RTT(config-object-group-network)# exit
RTT(config)# object-group network WAN_GATEWAY
RTT(config-object-group-network)# ip address-range 192.168.23.3
RTT(config-object-group-network)# exit
```

Для пропуска трафика из зоны «LAN» в зону «WAN» создадим пару зон и добавим правило, разрешающее проходить ICMP-трафику от ПК1 к ПК2. Действие правил разрешается командой *enable*:

```
RTT(config)# security zone-pair LAN WAN
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match destination-address WAN_GATEWAY
RTT(config-zone-pair-rule)# match source-address LAN_GATEWAY
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

Для пропуска трафика из зоны «WAN» в зону «LAN» создадим пару зон и добавим правило, разрешающее проходить ICMP-трафику от ПК2 к ПК1. Действие правил разрешается командой *enable*:

```
RTT(config)# security zone-pair WAN LAN
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match destination-address LAN_GATEWAY
RTT(config-zone-pair-rule)# match source-address WAN_GATEWAY
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

На маршрутизаторе всегда существует зона безопасности с именем «self». Если в качестве получателя трафика выступает сам маршрутизатор, то есть трафик не является транзитным, то в качестве параметра указывается зона «self». Создадим пару зон для трафика, идущего из зоны «WAN» в зону «self». Добавим правило, разрешающее проходить ICMP-трафику между ПК2 и маршрутизатором RTT, для того чтобы маршрутизатор начал отвечать на ICMP-запросы из зоны «WAN»:

```
RTT(config)# security zone-pair WAN self
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match destination-address WAN
RTT(config-zone-pair-rule)# match source-address WAN_GATEWAY
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

Создадим пару зон для трафика, идущего из зоны «LAN» в зону «self». Добавим правило, разрешающее проходить ICMP-трафику между ПК1 и RTT, для того чтобы маршрутизатор начал отвечать на ICMP-запросы из зоны «LAN»:

```
RTT(config)# security zone-pair LAN self
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match destination-address LAN
RTT(config-zone-pair-rule)# match source-address LAN_GATEWAY
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# exit
```

Посмотреть членство портов в зонах можно с помощью команды:

```
RTT# show security zone
```

Посмотреть пары зон и их конфигурацию можно с помощью команд:

```
RTT# show security zone-pair
RTT# show security zone-pair configuration
```

Посмотреть активные сессии можно с помощью команд:

```
RTT# show ip firewall sessions
```


7.15 Настройка списков доступа (ACL)

Access Control List или ACL — список контроля доступа, содержит правила, определяющие прохождение трафика через интерфейс.

7.15.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать список контроля доступа и перейти в режим его конфигурирования.	<code>RTT(config)# ip access-list extended <NAME></code>	<NAME> – имя создаваемого списка контроля доступа, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого списка контроля доступа (не обязательно).	<code>RTT(config-acl)# description <DESCRIPTION></code>	<DESCRIPTION> – описание списка контроля доступа, задаётся строкой до 255 символов.
3	Создать правило и перейти в режим его конфигурирования. Правила обрабатываются маршрутизатором в порядке возрастания их номеров.	<code>RTT(config-acl)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1..4094].
4	Указать действие, которое должно быть применено для трафика, удовлетворяющего заданным критериям.	<code>RTT(config-acl-rule)# action <ACT></code>	<ACT> – назначаемое действие: permit – прохождение трафика разрешается; deny – прохождение трафика запрещается.
5	Установить имя/номер протокола, для которого должно срабатывать правило.	<code>RTT(config-acl-rule)# match protocol <TYPE></code>	<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;
		<code>RTT(config-acl-rule)# match protocol-id <ID></code>	<ID> – идентификационный номер IP-протокола, принимает значения [0x00-0xFF].
6	Установить IP-адреса отправителя, для которых должно срабатывать правило.	<code>RTT(config-acl-rule)# match source-address { <ADDR> <MASK> any }</code>	<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.
7	Установить IP-адреса получателя, для которых должно срабатывать правило.	<code>RTT(config-acl-rule)# match destination-address { <ADDR> <MASK> any }</code>	При указании значения «any» правило будет срабатывать для любого IP-адреса отправителя/получателя.

8	Установить MAC-адреса отправителя, для которых должно срабатывать правило (не обязательно).	RTT (config-acl-rule) # match source-mac <ADDR><WILDCARD>	<ADDR> – MAC-адрес отправителя, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF];
9	Установить MAC-адреса получателя, для которых должно срабатывать правило (не обязательно).	RTT (config-acl-rule) # match destination-mac <ADDR><WILDCARD>	<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.
10	Установить номер TCP/UDP-порта отправителя, для которого должно срабатывать правило (если указан протокол).	RTT (config-acl-rule) # match source-port { <PORT> any }	<PORT> – номер TCP/UDP-порта отправителя, принимает значения [1..65535]. При указании значения «any» правило будет срабатывать для любого TCP/UDP-порта отправителя.
11	Установить номер TCP/UDP-порта получателя, для которого должно срабатывать правило (если указан протокол).	RTT (config-acl-rule) # match destination-port { <PORT> any }	
12	Установить значение 802.1p приоритета, для которого должно срабатывать правило (не обязательно).	RTT (config-acl-rule) # match cos <COS>	<COS> – значение 802.1p приоритета, принимает значения [0..7].
13	Установить значение кода DSCP, для которого должно срабатывать правило (не обязательно). Невозможно использовать совместно с IP Precedence.	RTT (config-acl-rule) # match dscp <DSCP>	<DSCP> – значение кода DSCP, принимает значения [0..63].
14	Установить значение кода IP Precedence, для которого должно срабатывать правило (не обязательно). Невозможно использовать совместно с DSCP.	RTT (config-acl-rule) # match ip-precedence <IPP>	<IPP> – значение кода IP Precedence, принимает значения [0..7].
15	Установить значение идентификационного номера VLAN, для которого должно срабатывать правило (не обязательно).	RTT (config-acl-rule) # match vlan <VID>	<VID> – идентификационный номер VLAN, принимает значения [1..4094].
16	Активировать правило.	RTT (config-acl-rule) # enable	

17	Указать список контроля доступа к конфигурируемому интерфейсу для фильтрации входящего трафика.	RTT(config-if-gi)# service-acl input <NAME>	<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.
----	---	---	---

Также списки доступа могут использоваться для организации политик QoS.

7.15.2 Пример настройки списка доступа

Задача:

Разрешить прохождения трафика только из подсети 192.168.20.0/24.

Решение:

Настроим список доступа для фильтрации по подсетям:

```
RTT# configure
RTT(config)# ip access-list extended white
RTT(config-acl)# rule 1
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address 192.168.20.0 255.255.255.0
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
```

Применим список доступа на интерфейс Gi1/0/19 для входящего трафика:

```
RTT(config)# interface gigabitethernet 1/0/19
RTT(config-if-gi)# service-acl input white
```

Просмотреть детальную информацию о списке доступа возможно через команду:

```
RTT# show ip access-list white
```

7.16 Конфигурирование статических маршрутов

Статическая маршрутизация – вид маршрутизации, при котором маршруты указываются в явном виде при конфигурации маршрутизатора без использования протоколов динамической маршрутизации.

7.16.1 Процесс настройки

Добавить статический маршрут возможно командой в режиме глобальной конфигурации:

```
RTT(config)# ip route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> | interface <IF> | tunnel  
<TUN> | wan load-balance rule <RULE> [<METRIC>] | blackhole | unreachable |  
prohibit } [ <METRIC> ] [ track <TRACK-ID> ] [ bfd ]
```

- <VRF> – имя экземпляра VRF, задается строкой до 31 символа;
- <SUBNET> – адрес назначения, может быть задан в следующем формате:
 - 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];
- <IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе 3.3;
- <TUN> – имя туннеля, задаётся в виде, описанном в разделе 0;
- <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);
- bfd – при указании данного ключа активируется удаление статического маршрута в случае недоступности next-hop.

Для добавления статического IPv6-маршрута к указанной подсети используется команда:

```
ipv6 route [ vrf <VRF> ] <SUBNET> { <NEXTHOP> [ resolve ] | interface <IF> | wan  
load-balance rule <RULE> | blackhole | unreachable | prohibit } [ <METRIC> ]  
[ bfd ]
```

- <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;
- <IF> – имя IP-интерфейса, задаётся в виде, описанном в разделе 3.3;
- 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].
- bfd – при указании данного ключа активируется удаление статического маршрута в случае недоступности next-hop.

7.16.2 Пример настройки статических маршрутов

Задача:

Настроить доступ к сети Internet для пользователей локальных сетей 192.168.1.0/24 и 10.0.0.0/8, используя статическую маршрутизацию. На устройстве R1 создать шлюз для доступа к сети Internet. Трафик внутри локальной сети должен маршрутизироваться внутри зоны LAN, трафик из сети Internet должен относиться к зоне WAN.

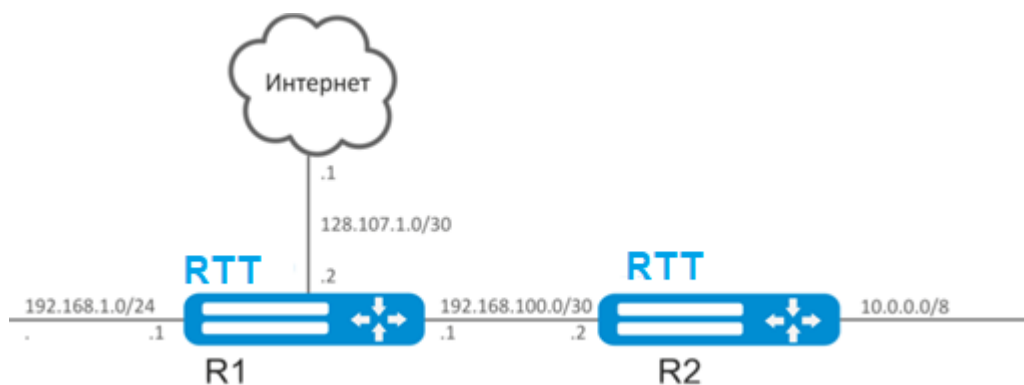


Рисунок 33 – Схема сети

Решение:

Зададим имя устройства для маршрутизатора R1:

```
RTT# hostname R1
```

Для интерфейса gi1/0/1 укажем адрес 192.168.1.1/24 и зону «LAN». Через данный интерфейс R1 будет подключен к сети 192.168.1.0/24:

```
RTT(config)# interface gi1/0/1
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# ip address 192.168.1.1/24
RTT(config-if-gi)# exit
```

Для интерфейса gi1/0/2 укажем адрес 192.168.100.1/30 и зону «LAN». Через данный интерфейс R1 будет подключен к устройству R2 для последующей маршрутизации трафика:

```
RTT(config)# interface gi1/0/2
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# ip address 192.168.100.1/30
RTT(config-if-gi)# exit
```

Для интерфейса gi1/0/3 укажем адрес 128.107.1.2/30 и зону «WAN». Через данный интерфейс R1 будет подключен к сети Internet:

```
RTT(config)# interface gi1/0/3
RTT(config-if-gi)# security-zone WAN
RTT(config-if-gi)# ip address 128.107.1.2/30
RTT(config-if-gi)# exit
```

Создадим маршрут для взаимодействия с сетью 10.0.0.0/8, используя в качестве шлюза устройство R2 (192.168.100.2):

```
RTT(config)# ip route 10.0.0.0/8 192.168.100.2
```

Создадим маршрут для взаимодействия с сетью Internet, используя в качестве nexthop шлюз провайдера (128.107.1.1):

```
RTT(config)# ip route 0.0.0.0/0 128.107.1.1
```

Зададим имя устройства для маршрутизатора R2:

```
RTT# hostname R2
```

Для интерфейса gi1/0/1 укажем адрес 10.0.0.1/8 и зону «LAN». Через данный интерфейс R2 будет подключен к сети 10.0.0.0/8:

```
RTT(config)# interface gi1/0/1
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# ip address 10.0.0.1/8
RTT(config-if-gi)# exit
```

Для интерфейса gi1/0/2 укажем адрес 192.168.100.2/30 и зону «LAN». Через данный интерфейс R2 будет подключен к устройству R1 для последующей маршрутизации трафика:

```
RTT(config)# interface gi1/0/2
RTT(config-if-gi)# security-zone LAN
RTT(config-if-gi)# ip address 192.168.100.2/30
RTT(config-if-gi)# exit
```

Создадим маршрут по умолчанию, указав в качестве nexthop IP-адрес интерфейса gi1/0/2 маршрутизатора R1 (192.168.100.1):

```
RTT(config)# ip route 0.0.0.0/0 192.168.100.1
```

Проверить таблицу маршрутов можно командой:

```
RTT# show ip route
```

7.17 Настройка MLPPP

Multilink PPP (MLPPP) предоставляет собой агрегированный канал, включающий в себя методы для распространения трафика через несколько физических каналов, имея одно логическое соединение. Этот вариант позволяет расширить пропускную способность и обеспечивает балансировку нагрузки.



Рисунок 34 – Схема сети

7.17.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить группу агрегации.	<code>RTT(config)# interface multilink <IF></code>	<IF>– наименование интерфейса.
2	Указать описание конфигурируемой группы агрегации (не обязательно).	<code>RTT(config- multilink)# description <DESCRIPTION></code>	<DESCRIPTION> – описание группы агрегации, задаётся строкой до 255 символов.
3	Задать интервал времени, за который усредняется статистика о нагрузке на группе агрегации (не обязательно).	<code>RTT(config- multilink)# load- average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.
4	Указать размер MTU (Maximum Transmission Unit) для группы агрегации (не обязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames".	<code>RTT(config- multilink)# mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [1280..1500]. Значение по умолчанию: 1500.
6	Включить CHAP-аутентификацию.	<code>RTT(config- multilink)# ppp authentication chap</code>	
7	Включить игнорирование аутентификации (не обязательно).	<code>RTT(config- multilink)# ppp chap refuse</code>	
8	Указать имя маршрутизатора, которое отправляется удаленной стороне для прохождения CHAP-аутентификации.	<code>RTT(config- multilink)# ppp chap hostname <NAME></code>	<NAME> – имя маршрутизатора, задаётся строкой до 31 символа.
9	Указать пароль, который отправляется удаленной стороне вместе с именем маршрутизатора для прохождения CHAP-аутентификации.	<code>RTT(config- multilink)# ppp chap password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED- TEXT> }</code>	<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F]. <ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.
10	Разрешить принимать от	<code>RTT(config-</code>	

	соседа любой ненулевой IP-адрес в качестве локального IP-адреса (не обязательно).	<code>multilink)# ppp ipcp accept-address</code>	
11	Установить IP-адрес, который отправляется удаленной стороне для последующего его присвоения.	<code>RTT(config-multilink)# ppp iccp remote-address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.
12	Указать пользователя для аутентификации удаленной стороны и перейти в режим конфигурирования указанного пользователя	<code>RTT(config-multilink)# chap username <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
13	Установить пароль в открытой или зашифрованной форме определенному пользователю для аутентификации удаленной стороны.	<code>RTT(config-ppp-user)# password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<CLEAR-TEXT> – пароль в открытой форме, задаётся строкой [8 .. 64] символов, может включать символы [0-9a-fA-F]. <ENCRYPTED-TEXT> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.
14	Установить количество попыток отправки Configure-Request пакетов, прежде чем удаленный пир будет признан неспособным ответить (не обязательно).	<code>RTT(config-multilink)# ppp max-configure <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 10.
15	Установить количество попыток выслать Configure-NAK пакеты, прежде чем будут подтверждены все опции (не обязательно).	<code>RTT(config-multilink)# ppp max-failure <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255].
16	Установить количество попыток выслать Terminate-Request пакеты, прежде чем сессия будет прервана (не обязательно).	<code>RTT(config-multilink)# ppp max-terminate <VALUE></code>	<VALUE> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 2.
17	Указать размер MRU (Maximum Receive Unit) для интерфейса.	<code>RTT(config-multilink)# ppp mru <MRU></code>	<MRU> – значение MRU, принимает значения в диапазоне [128..1485]. Значение по умолчанию: 1500.

18	Указать интервал времени в секундах, по истечении которого маршрутизатор отправляет keepalive-сообщение (не обязательно).	<code>RTT(config-multilink)# ppp timeout keepalive <TIME></code>	<TIME> – время в секундах, принимает значения [1..32767]. Значение по умолчанию: 10.
19	Установить интервал времени в секундах, по истечении которого маршрутизатор повторяет запрос на установление сессии (не обязательно).	<code>RTT(config-multilink)# ppp timeout retry <TIME></code>	<TIME> – время в секундах, принимает значения [1..255]. Значение по умолчанию: 3.
20	Определить максимальный размер пакета для MLPP интерфейса.	<code>RTT(config-multilink)# mrru <MRRU></code>	<MRRU> – максимальный размер принимаемого пакета для MLPP интерфейса, принимает значение в диапазоне [1500..10000].
21	Привязать порт e1 к физическому интерфейсу.	<code>RTT(config-if-gi)# switchport e1 <SLOT></code>	<SLOT> – идентификатор слота, принимает значение в диапазоне [0..3].
22	Перевести физический порт в режим работы с SFPe1 модулем.	<code>RTT(config-if-gi)# switchport mode e1</code>	
23	Включить режим MLPPP на E1-интерфейсе.	<code>RTT(config-e1)# ppp multilink</code>	
24	Включить E1-интерфейс в группу агрегации.	<code>RTT(config-e1)# ppp multilink-group <GROUP-ID></code>	<GROUP-ID> – идентификатор группы, принимает значение [1..4].

7.17.2 Пример настройки

Задача:

Настроить MLPPP-соединение с встречной стороной с IP-адресом 10.77.0.1/24 через устройство MXE.



Рисунок 35 – Схема сети

Решение:

Переключаем интерфейс gigabitethernet 1/0/10 в режим работы E1:

```

RTT# configure
RTT(config)# interface gigabitethernet 1/0/10
RTT(config-if-gi)# description "MXE"
RTT(config-if-gi)# switchport mode e1
RTT(config-if-gi)# switchport e1 slot 0
RTT(config-if-gi)# exit
  
```


Настроим MLPPP 3:

```
RTT(config)# interface multilink 3
RTT(config-multilink)# ip address 10.77.0.2/24
RTT(config-multilink)# security-zone trusted
RTT(config-multilink)# exit
RTT(config)# exit
```

Включим interface e1 1/0/1, interface e1 1/0/4 в группу агрегации MLPPP 3:

```
RTT(config)# interface e1 1/0/1
RTT(config-e1)# ppp multilink
RTT(config-e1)# ppp multilink-group 3
RTT(config-e1)# exit
RTT(config)# interface e1 1/0/4
RTT(config-e1)# ppp multilink
RTT(config-e1)# ppp multilink-group 3
RTT(config-e1)# exit
```

7.18 Настройка Bridge

Bridge (мост) — это способ соединения двух сегментов Ethernet на канальном уровне без использования протоколов более высокого уровня, таких как IP. Пакеты передаются на основе Ethernet-адресов, а не IP-адресов. Поскольку передача выполняется на канальном уровне (уровень 2 модели OSI), трафик протоколов более высокого уровня прозрачно проходит через мост.

7.18.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Добавить сетевой мост (bridge) в систему и перейти в режим настройки его параметров.	RTT(config)# bridge <BRIDGE-ID>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: для RTT-RTT-R100/RTT-R200 – [1..250], для RTT-RTT-R800 - [1..500].
2	Активировать сетевой мост.	RTT(config-bridge)# enable	
3	Указать экземпляр VRF, в котором будет работать данный интерфейс (не обязательно).	RTT(config-bridge)# ip vrf forwarding <VRF>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Назначить описание конфигурируемому сетевому мосту.	RTT(config-bridge)# description <DESCRIPTION>	<DESCRIPTION> – описание сетевого моста, задаётся строкой до 255 символов.
5	Связать текущий сетевой мост с VLAN. Все порты, являющиеся членами назначаемого VLAN, автоматически включаются в сетевой мост и становятся участниками общего L2 домена	RTT(config-bridge)# vlan <VID>	<VID> – идентификатор VLAN, задаётся в диапазоне [1..4094].

6	Задать MAC-адрес сетевого моста, отличный от системного (не обязательно).	RTT (config-bridge) # mac-address <ADDR>	<ADDR> – MAC-адрес сетевого моста, задаётся в виде XX:XX:XX:XX:XX:XX, где каждая часть принимает значения [00..FF].
7	Связать саб-интерфейс, QinQ-интерфейс, L2GRE туннель или L2TPv3 туннель с сетевым мостом. Связанные интерфейсы/туннели и сетевые мосты автоматически становятся участниками общего L2 домена.	RTT (config-if-gi) # bridge-group <BRIDGE-ID> RTT (config-if-l2tpv3) # bridge-group <BRIDGE-ID>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: RTT-R100/200 – [1..250], для RTT-RTT-R800 – [1..500]

7.18.2 Пример настройки bridge для VLAN и L2TPv3-туннеля

Задача:

Объединить в единый L2 домен интерфейсы маршрутизатора, относящиеся к локальной сети, и L2TPv3-туннель, проходящий по публичной сети. Для объединения использовать VLAN 333.

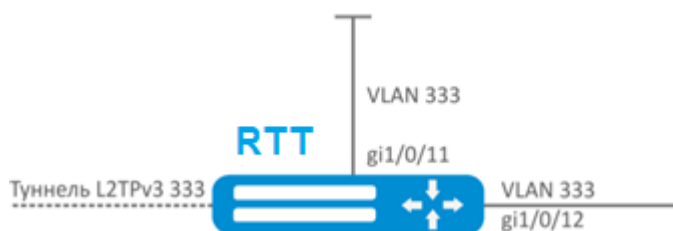


Рисунок 36 – Схема сети

Решение:

Создадим VLAN 333:

```
RTT(config)# vlan 333
RTT(config-vlan)# exit
```

Создадим зону безопасности «trusted»:

```
RTT(config)# security-zone trusted
RTT(config-zone)# exit
```

Добавим интерфейсы gi1/0/11, gi1/0/12 в VLAN 333:

```
RTT(config)# interface gigabitethernet 1/0/11-12
RTT(config-if)# switchport general allowed vlan add 333 tagged
```

Создадим bridge 333, привяжем к нему VLAN 333 и укажем членство в зоне «trusted»:

```
RTT(config)# bridge 333
RTT(config-bridge)# vlan 333
RTT(config-bridge)# security-zone trusted
```

```
RTT(config-bridge)# enable
```

Установим принадлежность L2TPv3-туннеля к мосту, который связан с локальной сетью (настройка L2TPv3-туннеля рассматривается в разделе 7.25). В общем случае идентификаторы моста и туннеля не должны совпадать с VID как в данном примере.

```
RTT(config)# tunnel l2tpv3 333
RTT(config-l2tpv3)# bridge-group 333
```

7.18.3 Пример настройки bridge для VLAN

Задача:

Настроить маршрутизацию между VLAN 50 (10.0.50.0/24) и VLAN 60 (10.0.60.0/24). VLAN 50 должен относиться к зоне «LAN1», VLAN 60 – к зоне «LAN2», разрешить свободную передачу трафика между зонами.

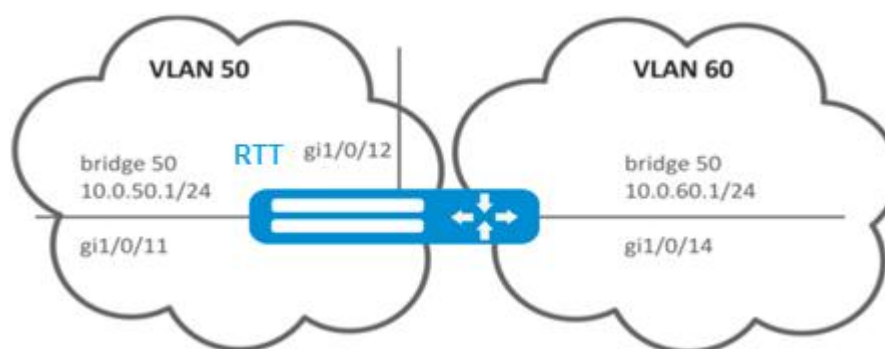


Рисунок 37 – Схема сети

Решение:

Создадим VLAN 50, 60:

```
RTT(config)# vlan 50,60
RTT(config-vlan)# exit
```

Создадим зоны безопасности «LAN1» и «LAN2»:

```
RTT(config)# security-zone LAN1
RTT(config-zone)# exit
RTT(config)# security-zone LAN2
RTT(config-zone)# exit
```

Назначим интерфейсам gi1/0/11, gi1/0/12 VLAN 50:

```
RTT(config)# interface gigabitethernet 1/0/11-12
RTT(config-if-gi)# switchport general allowed vlan add 50 tagged
```

Назначим интерфейсу gi1/0/14 VLAN 60:

```
RTT(config)# interface gigabitethernet 1/0/14
RTT(config-if-gi)# switchport general allowed vlan add 60 tagged
```

Создадим bridge 50, привяжем VLAN 50, укажем IP-адрес 10.0.50.1/24 и членство в зоне «LAN1»:

```
RTT(config)# bridge 50
RTT(config-bridge)# vlan 50
RTT(config-bridge)# ip address 10.0.50.1/24
RTT(config-bridge)# security-zone LAN1
RTT(config-bridge)# enable
```

Создадим bridge 60, привяжем VLAN 60, укажем IP-адрес 10.0.60.1/24 и членство в зоне «LAN2»:

```
RTT(config)# bridge 60
RTT(config-bridge)# vlan 60
RTT(config-bridge)# ip address 10.0.60.1/24
RTT(config-bridge)# security-zone LAN2
RTT(config-bridge)# enable
```

Создадим правила в Firewall, разрешающие свободное прохождение трафика между зонами:

```
RTT(config)# security zone-pair LAN1 LAN2
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# security zone-pair LAN2 LAN1
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# exit
```

Посмотреть членство интерфейсов в мосте можно командой:

```
RTT# show interfaces bridge
```

7.18.4 Пример настройки добавления/удаления второго VLAN-тега

Задача:

На интерфейс gigabitethernet 1/0/1 поступают Ethernet-кадры с различными VLAN-тегами. Необходимо перенаправить их в интерфейс gigabitethernet 1/0/2, добавив второй VLAN-ID 828. При поступлении на интерфейс gigabitethernet 1/0/2 Ethernet-кадров с VLAN-ID 828, данный тег должен быть удален и отправлен в интерфейс gigabitethernet 1/0/1.

Решение:

Создадим на маршрутизаторе bridge без VLAN и без IP-адреса.

```
RTT(config)# bridge 1
RTT(config-bridge)# enable
RTT(config-bridge)# exit
```

Включим интерфейс gigabitethernet 1/0/1 в bridge 1.

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# bridge-group 1
RTT(config-if-gi)# exit
```

Включим саб-интерфейс gigabitethernet 1/0/2.828 в bridge 1.

```
RTT(config)# interface gigabitethernet 1/0/2.828
RTT(config-subif)# bridge-group 1
RTT(config-subif)# exit
```



При добавлении второго VLAN-тэга в Ethernet-кадр, его размер увеличивается на 4 байта. На интерфейсе маршрутизатора gigabitethernet 1/0/2 и на всем оборудовании передающем q-in-q-кадры необходимо увеличить MTU на 4 байта или более.

7.19 Настройка RIP

RIP — дистанционно-векторный протокол динамической маршрутизации, который использует количество транзитных участков в качестве метрики маршрута. Максимальное количество транзитных участков (hop), разрешенное в RIP, равно 15. Каждый RIP-маршрутизатор по умолчанию вещает в сеть свою полную таблицу маршрутизации один раз в 30 секунд. RIP работает на 3-м уровне стека TCP/IP, используя UDP-порт 520.

7.19.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола RIP маршрутизации для основной таблицы маршрутизации (не обязательно).	<code>RTT(config)# ip protocols rip preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: RIP (100).
2	Настроить емкость таблиц маршрутизации протокола RIP (не обязательно).	<code>RTT(config)# ip protocols rip max-routes <VALUE></code>	<VALUE> – количество маршрутов протокола RIP в маршрутной таблице, принимает значения в диапазоне: [1..10000] Значение по умолчанию: (10000),
3	Создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<code>RTT(config)# ip prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.

4	Разрешить (permit) или запретить (deny) списки префиксов.	<pre>RTT(config-pl)# permit {object-group <OBJ-GROUP-NETWORK- NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default- route} RTT(config-pl)# deny {object-group <OBJ- GROUP-NETWORK-NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default-route}</pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP -адресов, задаётся строкой до 31 символа;</p> <p><LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов;</p> <p>eq – при указании команды длина префикса должна соответствовать указанной;</p> <p>le – при указании команды длина префикса должна быть меньше либо соответствовать указанной;</p> <p>ge – при указании команды длина префикса должна быть больше либо соответствовать указанной;</p> <p>default-route – фильтрация маршрута по умолчанию.</p>
5	Перейти в режим настройки параметров RIP-процесса.	<pre>RTT(config)# router rip RTT(config-rip)#</pre>	
6	Включить RIP-протокол.	<pre>RTT(config-rip)# enable</pre>	
7	Определить алгоритм аутентификации протокола RIP (не обязательно).	<pre>RTT(config-rip)# authentication algorithm { cleartext md5 }</pre>	<p>cleartext – пароль, передается открытым текстом;</p> <p>md5 – пароль хешируется по алгоритму md5.</p>
8	Установить пароль для аутентификации с соседом (не обязательно).	<pre>RTT(config-rip)# authentication key ascii-text { <CLEAR- TEXT> encrypted <ENCRYPTED-TEXT> }</pre>	<p><CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов;</p> <p><ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).</p>
9	Определить список паролей для аутентификации через алгоритм хеширования md5 (не обязательно).	<pre>RTT(config-rip)# authentication key- chain <KEYCHAIN></pre>	<p><KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.</p>
10	Выключить анонсирование маршрутов на интерфейсах/туннелях/bridge, где это не нужно (не обязательно).	<pre>RTT(config-rip)# passive-interface {<IF> <TUN> }</pre>	<p><IF> – интерфейс и идентификатор;</p> <p><TUN> – имя и номер туннеля.</p>
11	Установить временной интервал, по истечении которого производится анонсирование (не обязательно).	<pre>RTT(config-rip)# timers update <TIME></pre>	<p><TIME> – время в секундах, принимает значения [1..65535].</p> <p>Значение по умолчанию: 180 секунд.</p>

12	Установить временной интервал корректности маршрутной записи без обновления (не обязательно).	RTT(config-rip) # timers invalid <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 180 секунд.
13	Установить временной интервал, по истечении которого производится удаление маршрута (не обязательно).	RTT(config-rip) # timers flush <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. При установке значения нужно учитывать следующее правило: «timersinvalid + 60» Значение по умолчанию: 240 секунд.
14	Включить анонсирование подсетей.	RTT(config-rip) # network <ADDR/LEN>	<ADDR/LEN> – адрес подсети, указывается в следующем формате: 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].
15	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (не обязательно).	RTT(config-rip) # prefix-list <PREFIX- LIST-NAME> { in out }	<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа. in – фильтрация входящих маршрутов; out – фильтрация анонсируемых маршрутов.
16	Включить анонсирование маршрутов, полученных альтернативным способом (не обязательно).	RTT(config-rip) # redistribute static [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.
		RTT(config-rip) # redistribute connected [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных подсетей, задаётся строкой до 31 символа.
		RTT(config-rip) # redistribute ospf <ID><ROUTE-TYPE> [route-map <NAME>]	<ID> – номер процесса, может принимать значение [1..65535]; <ROUTE-TYPE> – тип маршрута: 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; <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.

		RTT(config-rip)# redistribute bgp <AS> [route-map <NAME>]	<AS> – номер автономной системы, может принимать значения [1..4294967295]; <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.
17	Перейти в режим конфигурирования интерфейса/туннеля/сетевого моста.	RTT(config)# interface <IF- TYPE><IF-NUM>	<IF-TYPE> тип интерфейса; <IF-NUM> - F/S/P – F-фрейм (1), S – слот (0), P – порт.
		RTT(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> тип туннеля; <TUN-NUM> номер туннеля.
		RTT(config)# bridge <BR-NUM>	<BR-NUM> - номер bridge.
18	Установить величину метрики RIP-маршрутов на интерфейсе (не обязательно).	RTT(config-if-gi)# ip rip metric <VALUE>	<VALUE> – величина метрики, задаётся в размере [0..32767]. Значение по умолчанию: 5.
19	Установить режим анонсирования маршрутов по протоколу RIP (не обязательно).	RTT(config-if-gi)# ip rip mode <MODE>	<MODE> – режим анонсирования маршрутов: multicast – маршруты анонсируются в многоадресном режиме; broadcast – маршруты анонсируются в широковещательном режиме; unicast – маршруты анонсируются в unicast-режиме соседям. Значение по умолчанию: multicast.
20	Задать IP-адрес соседа для установления отношения в unicast-режиме анонсирования маршрутов (не обязательно).	RTT(config-if-gi)# ip rip neighbor <ADDR>	<ADDR> – IP-адрес, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
21	Включить суммаризацию подсетей (не обязательно).	RTT(config-if-gi)# ip rip summary-address <ADDR/LEN>	<ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

7.19.2 Пример настройки RIP

Задача:

Настроить на маршрутизаторе протокол RIP для обмена маршрутной информацией с соседними маршрутизаторами. Маршрутизатор должен анонсировать статические маршруты и подсети 115.0.0.0/24, 14.0.0.0/24, 10.0.0.0/24. Анонсирование маршрутов должно происходить каждые 25 секунд.

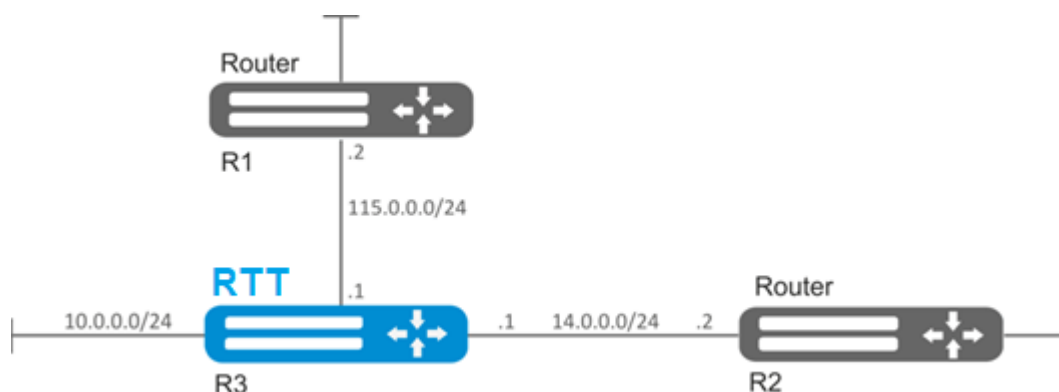


Рисунок 38 – Схема сети

Решение:

Предварительно нужно настроить IP-адреса на интерфейсах согласно схеме сети, приведенной на рисунке 38.

Перейдём в режим конфигурирования протокола RIP:

```
RTT(config)# router rip
```

Укажем подсети, которые будут анонсироваться протоколом: 115.0.0.0/24, 14.0.0.0/24 и 10.0.0.0/24:

```
RTT(config-rip)# network 115.0.0.0/24
RTT(config-rip)# network 14.0.0.0/24
RTT(config-rip)# network 10.0.0.0/24
```

Для анонсирования протоколом статических маршрутов выполним команду:

```
RTT(config-rip)# redistribute static
```

Настроим таймер, отвечающий за отправку маршрутной информации:

```
RTT(config-rip)# timers update 25
```

После установки всех требуемых настроек включаем протокол:

```
RTT(config-rip)# enable
```

Для того чтобы просмотреть таблицу маршрутов RIP воспользуемся командой:

```
RTT# show ip rip
```



Помимо настройки протокола RIP, необходимо в firewall разрешить UDP-порт 520.

7.20 Настройка OSPF

OSPF — протокол динамической маршрутизации, основанный на технологии отслеживания состояния канала (link-state technology) и использующий для нахождения кратчайшего пути алгоритм Дейкстры.

7.20.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола OSPF маршрутизации для основной таблицы маршрутизации (не обязательно).	<code>RTT(config)# ip protocols ospf preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: 150.
		<code>RTT(config-vrf)# ip protocols ospf preference <VALUE></code>	
2	Настроить емкость таблиц маршрутизации протокола OSPF (не обязательно).	<code>RTT(config)# ip protocols ospf max-routes <VALUE></code>	<VALUE> – количество маршрутов протокола OSPF в маршрутной таблице, принимает значения в диапазоне: для RTT-R800 [1..500000]; для RTT-R100/200 [1..300000]. Значение по умолчанию для глобального режима: для RTT-R800 – (500000); для RTT-R100/200 – (300000); Значение по умолчанию для глобального режима: 0
		<code>RTT(config)# ipv6 protocols ospf max-routes <VALUE></code>	
3	Включить вывод информации о состоянии отношений с соседями для протокола маршрутизации OSPF (не обязательно).	<code>RTT(config)# router ospf log-adjacency-changes</code>	
		<code>RTT(config)# ipv6 router ospf log-adjacency-changes</code>	
4	Создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<code>RTT(config)# ip prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
		<code>RTT(config)# ipv6 prefix-list <NAME></code>	
5	Разрешить (permit) или запретить (deny) списки префиксов.	<code>RTT(config-pl)# permit {object-group <OBJ-GROUP-NETWORK-NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default-route}</code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа; <LEN> – длина префикса, принимает значения [1..32] в IP-списках префиксов; eq – при указании команды длина префикса должна соответствовать указанной; le – при указании команды длина префикса должна быть меньше либо
		<code>RTT(config-pl)# deny {object-group <OBJ-GROUP-NETWORK-NAME> [{ eq <LEN> le</code>	

		<pre><LEN> ge <LEN> [le <LEN>] }] default-route} RTT(config-ipv6-pl)# permit {object-group <OBJ-GROUP-NETWORK- NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default- route} RTT(config-ipv6-pl)# deny object-group <OBJ-GROUP-NETWORK- NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default-route}</pre>	<p>соответствовать указанной; ge – при указании команды длина префикса должна быть больше либо соответствовать указанной; default-route – фильтрация маршрута по умолчанию.</p>
6	Добавить OSPF-процесс в систему и осуществить переход в режим настройки параметров OSPF-процесса.	<pre>RTT(config)# router ospf <ID> [vrf <VRF>] RTT(config)# ipv6 router ospf <ID> [vrf <VRF>]</pre>	<p><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.</p>
7	Установить идентификатор маршрутизатора для данного OSPF-процесса.	<pre>RTT(config-ospf)# router-id <ID> RTT(config-ipv6- ospf)# router-id <ID></pre>	<p><ID> – идентификатор маршрутизатора, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
8	Определить приоритетность маршрутов процесса OSPF.	<pre>RTT(config-ospf)# preference <VALUE> RTT(config-ipv6- ospf)# preference <VALUE></pre>	<p><VALUE> – приоритетность маршрутов процесса OSPF, принимает значения в диапазоне [1..255]. Значение по умолчанию: 10.</p>
9	Включить совместимость с RFC 1583 (не обязательно).	<pre>RTT(config-ospf)# compatible rfc1583 RTT(config-ipv6- ospf)# compatible rfc1583</pre>	
11	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (не обязательно).	<pre>RTT(config-ospf)# prefix-list <PREFIX- LIST-NAME> { in out } RTT(config-ipv6- ospf)# prefix-list <PREFIX-LIST-NAME> { in out }</pre>	<p><PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задается строкой до 31 символа. in – фильтрация входящих маршрутов; out – фильтрация анонсируемых маршрутов.</p>
12	Включить анонсирование маршрутов, полученных альтернативным способом (не обязательно).	<pre>RTT(config-ospf)# redistribute static [route-map <NAME>] RTT(config-ipv6- ospf)# redistribute static [route-map <NAME>] RTT(config-ospf)# redistribute connected [route-map <NAME>] RTT(config-ipv6-</pre>	<p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задается строкой до 31 символа.</p> <p><NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключенных</p>

		<code>ospf)# redistribute connected [route-map <NAME>]</code>	подсетей, задаётся строкой до 31 символа.
		<code>RTT(config-ospf)# redistribute rip [route-map <NAME>]</code>	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.
		<code>RTT(config-ospf)# redistribute bgp <AS> [route-map <NAME>]</code>	<AS> – номер автономной системы, может принимать значения [1..4294967295]; <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.
		<code>RTT(config-ipv6-ospf)# redistribute bgp <AS> [route-map <NAME>]</code>	
13	Активировать OSPF-процесс.	<code>RTT(config-ospf)# enable</code>	
		<code>RTT(config-ipv6-ospf)# enable</code>	
14	Создать OSPF-область и перейти в режим конфигурирования области.	<code>RTT(config-ospf)# area <AREA_ID></code>	<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT(config-ipv6-ospf)# area <AREA_ID></code>	
15	Включить анонсирование подсетей.	<code>RTT(config-ospf-area)# network <ADDR/LEN></code>	<ADDR/LEN> – адрес подсети, указывается в следующем формате: 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].
		<code>RTT(config-ipv6-ospf-area)# network <IPV6-ADDR/LEN></code>	<IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128].
16	Определить тип области	<code>RTT(config-ospf-area)# area-type <TYPE> [no-summary]</code>	<TYPE> – тип области: stub – устанавливает значение stub (тупиковая область); no-summary – команда в связке с параметром «stub» образует область «totallystubby» (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 totallynssa (автоматически генерирует маршрут по умолчанию как межобластной).
		<code>RTT(config-ipv6-ospf-area)# area-type <TYPE> [no-summary]</code>	
17	Включить генерацию	<code>RTT(config-ospf-area)# default-</code>	

	маршрута по умолчанию для NSSA-области и анонсирование его в качестве NSSA-LSA.	<code>information-originate</code> <code>RTT(config-ipv6-ospf-area)# default-information-originate</code>	
18	Включить суммаризацию или скрытие подсетей.	<code>RTT(config-ospf-area)# summary-address <ADDR/LEN> { advertise not-advertise }</code> <code>RTT(config-ipv6-ospf-area)# summary-address <IPV6-ADDR/LEN> { advertise not-advertise }</code>	<p><ADDR/LEN> – IP-адрес и маска подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32];</p> <p>advertise – при указании команды вместо указанных подсетей будет анонсироваться суммарная подсеть;</p> <p>not-advertise – при указании команды подсети, входящие в указанную подсеть, анонсироваться не будут.</p> <p><IPV6-ADDR/LEN> – IPv6-адрес и маска подсети, задаётся в виде X:X:X:X/EE, где каждая часть X принимает значения в шестнадцатеричном формате [0..FFFF] и EE принимает значения [1..128];</p> <p>advertise – при указании команды вместо подсетей, входящих в указанную подсеть, будет анонсироваться суммарная подсеть;</p> <p>not-advertise – подсети входящие в указанную подсеть анонсироваться не будут.</p>
19	Активировать OSPF-область.	<code>RTT(config-ospf-area)# enable</code> <code>RTT(config-ipv6-ospf-area)# enable</code>	
20	Установить виртуальное соединение между основной и удаленными областями, имеющими между ними несколько областей.	<code>RTT(config-ospf-area)# virtual-link <ID></code> <code>RTT(config-ipv6-ospf-area)# virtual-link <ID></code>	<p><ID> – идентификатор маршрутизатора, с которым устанавливается виртуальное соединение, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
21	Установить интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, который не получил подтверждения о получении (например, DatabaseDescription пакет или LinkStateRequest пакеты).	<code>RTT(config-ospf-vlink)# retransmit-interval <TIME></code> <code>RTT(config-ipv6-ospf-vlink)# retransmit-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [1..65535].</p> <p>Значение по умолчанию: 5 секунд.</p>
22	Установить интервал времени в секундах, по	<code>RTT(config-ospf-vlink)# hello-interval <TIME></code>	<p><TIME> – время в секундах, принимает значения [1..65535].</p>

	истечении которого маршрутизатор отправляет следующий hello-пакет.	RTT (config-ipv6-ospf-vlink) # hello-interval <TIME>	Значение по умолчанию: 10 секунд.
23	Установить интервал времени в секундах, по истечении которого сосед будет считаться неактивным. Этот интервал должен быть кратным значению «hello-interval».	RTT (config-ospf-vlink) # dead-interval <TIME> RTT (config-ipv6-ospf-vlink) # dead-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд.
24	Определяется интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети	RTT (config-ospf-vlink) # wait-interval <TIME> RTT (config-ipv6-ospf-vlink) # wait-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд
25	Определить алгоритм аутентификации	RTT (config-ospf-vlink) # authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации: cleartext – пароль, передается открытым текстом (доступно только для RIP и OSPF-VLINK); md5 – пароль хешируется по алгоритму md5.
26	Установить пароль для аутентификации с соседом.	RTT (config-ospf-vlink) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<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...).
27	Определить список паролей для аутентификации через алгоритм хеширования md5.	RTT (config-ospf-vlink) # authentication key chain <KEYCHAIN>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
28	Активировать виртуальное соединение.	RTT (config-ospf-vlink) # enable	
29	Перейти в режим конфигурирования интерфейса/туннеля/сетевого моста.	RTT (config) # interface <IF-TYPE><IF-NUM> RTT (config) # tunnel <TUN-TYPE><TUN-NUM> RTT (config) # bridge <BR-NUM>	<IF-TYPE> тип интерфейса; <IF-NUM> - F/S/P – F-фрейм (1), S – слот (0), P – порт. <TUN-TYPE> тип туннеля; <TUN-NUM> номер туннеля. <BR-NUM> - номер bridge.
30	Определить принадлежность интерфейса/туннеля/сетевого моста к определенному OSPF-процессу.	RTT (config-if-gi) # ip ospf instance <ID> RTT (config-if-gi) # ipv6 ospf instance <ID>	<ID> – номер процесса, принимает значения [1..65535].

31	Определить принадлежность интерфейса к определенной области OSPF-процесса.	RTT(config-if-gi)# ip ospf area <AREA_ID>	<AREA_ID> – идентификатор области, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		RTT(config-if-gi)# ipv6 ospf area <AREA_ID>	
32	Включить маршрутизацию по протоколу OSPF на интерфейсе.	RTT(config-if-gi)# ip ospf	
		RTT(config-if-gi)# ipv6 ospf	
33	Включить режим, в котором OSPF-процесс будет игнорировать значение MTU интерфейса во входящих Database Description-пакетах.	RTT(config-if-gi)# ip ospf mtu-ignore	
		RTT(config-if-gi)# ipv6 ospf mtu-ignore	
34	Определить алгоритм аутентификации протокола OSPF.	RTT(config-if-gi)# ip ospf authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации: cleartext – пароль, передается открытым текстом; md5 – пароль хешируется по алгоритму md5.
35	Установить пароль для аутентификации с OSPF-соседом при передаче пароля открытым текстом.	RTT(config-if-gi)# ip ospf authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<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...).
36	Определить список паролей для аутентификации по алгоритму хеширования md5 с соседом.	RTT(config-if-gi)# ip ospf authentication key-chain <KEYCHAIN>	<KEYCHAIN> – идентификатор списка ключей, задаётся строкой до 16 символов.
37	Определить интервал времени в секундах, по истечении которого маршрутизатор выберет DR в сети.	RTT(config-if-gi)# ip ospf wait-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд.
		RTT(config-if-gi)# ipv6 ospf wait-interval <TIME>	
38	Установить интервал времени в секундах, по истечении которого маршрутизатор повторно отправит пакет, на который не получил подтверждения о получении (например, DatabaseDescription пакет или LinkStateRequest пакеты).	RTT(config-if-gi)# ip ospf retransmit-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 5 секунд.
		RTT(config-if-gi)# ipv6 ospf retransmit-interval <TIME>	
39	Установить интервал времени в секундах, по	RTT(config-if-gi)# ip ospf hello-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535].

	истечении которого маршрутизатор отправляет следующий hello-пакет.	RTT(config-if-gi)# ipv6 ospf hello- interval <TIME>	Значение по умолчанию: 10 секунд.
40	Установить интервал времени в секундах, по истечении которого сосед будет считаться неактивным. Этот интервал должен быть кратным значению hello-interval.	RTT(config-if-gi)# ip dead-interval <TIME> RTT(config-if-gi)# ipv6 dead-interval <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 40 секунд.
41	Установить интервал времени, в течение которого NBMA-интерфейс ждет, прежде чем отправить HELLO-пакет соседу, даже в случае, если сосед неактивен.	RTT(config-if-gi)# ip poll-interval <TIME> RTT(config-if-gi)# ipv6 poll-interval <TIME>	<TIME> – время в секундах, принимает значения [1 .. 65535]. Значение по умолчанию: 120 секунд.
42	Задать статический IP-адрес соседа для установления отношения в NBMA и P2MP (Point-to-MultiPoint) сетях.	RTT(config-if-gi)# ip ospf neighbor <IP> [eligible] RTT(config-if-gi)# ip ospf neighbor <IP> [eligible]	<IP> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]. eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля. <IPV6-ADDR> – IPv6-адрес соседа, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]; eligible – опциональный параметр, позволяет устройству участвовать в процессе выбора DR в NBMA-сетях. Приоритет интерфейса должен быть больше нуля.
43	Определить тип сети для установления OSPF соседства.	RTT(config-if-gi)# ip ospf network <TYPE> RTT(config-if-gi)# ipv6 ospf network <TYPE>	<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.
44	Установить приоритет маршрутизатора, который используется для выбора	RTT(config-if-gi)# ip ospf priority <VALUE> RTT(config-if-gi)#	<VALUE> – приоритет интерфейса, принимает значения [1..65535]. Значение по умолчанию: 120.

	DR и BDR.	<code>ipv6 ospf priority <VALUE></code>	
45	Установить величину метрики на интерфейсе или туннеле.	<code>RTT(config-if-gi)# ip ospf cost <VALUE></code>	<VALUE> – величина метрики, задаётся в размере [0..32767]. Значение по умолчанию: 150.
		<code>RTT(config-if-gi)# ipv6 ospf cost <VALUE></code>	
47	Включить протокол BFD для протокола OSPF	<code>RTT(config-if-gi)# ip ospf bfd-enable</code>	
		<code>RTT(config-if-gi)# ipv6 ospf bfd-enable</code>	

7.20.2 Пример настройки OSPF

Задача:

Настроить протокол OSPF на маршрутизаторе для обмена маршрутной информацией с соседними маршрутизаторами. Маршрутизатор должен находиться в области с идентификатором 1.1.1.1 и анонсировать маршруты, полученные по протоколу RIP.

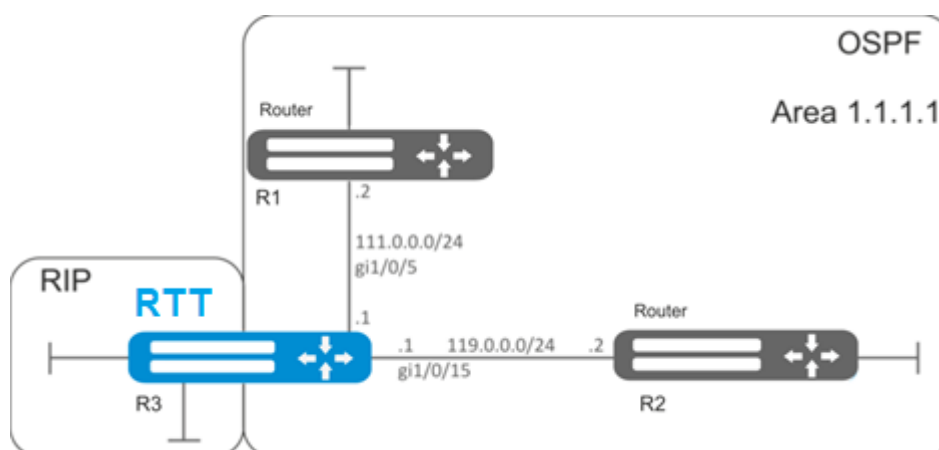


Рисунок 39 – Схема сети

Решение:

Предварительно нужно настроить IP-адреса на интерфейсах согласно схеме, приведенной на рисунке 39.

Создадим OSPF-процесс с идентификатором 10 и перейдём в режим конфигурирования протокола OSPF:

```
RTT(config) # router ospf 10
```

Создадим и включим требуемую область.

```
RTT(config-ospf) # area 1.1.1.1  
RTT(config-ospf-area) # enable  
RTT(config-ospf-area) # exit
```

Включим анонсирование маршрутной информации из протокола RIP:

```
RTT(config-ospf) # redistribute rip
```

Включим OSPF-процесс:

```
RTT(config-ospf) # enable  
RTT(config-ospf) # exit
```

Соседние маршрутизаторы подключены к интерфейсам gi1/0/5 и gi1/0/15. Для установления соседства с другими маршрутизаторами привяжем их к OSPF-процессу и области. Далее включим на интерфейсе маршрутизацию по протоколу OSPF:

```
RTT(config) # interface gigabitethernet 1/0/5  
RTT(config-if-gi) # ip ospf instance 10  
RTT(config-if-gi) # ip ospf area 1.1.1.1  
RTT(config-if-gi) # ip ospf  
RTT(config-if-gi) # exit
```

```
RTT(config) # interface gigabitethernet 1/0/15  
RTT(config-if-gi) # ip ospf instance 10  
RTT(config-if-gi) # ip ospf area 1.1.1.1  
RTT(config-if-gi) # ip ospf  
RTT(config-if-gi) # exit  
RTT(config) # exit
```

7.20.3 Пример настройки OSPF stub area

Задача:

Изменить тип области 1.1.1.1, область должна быть тупиковой. Тупиковый маршрутизатор должен анонсировать маршруты, полученные по протоколу RIP.

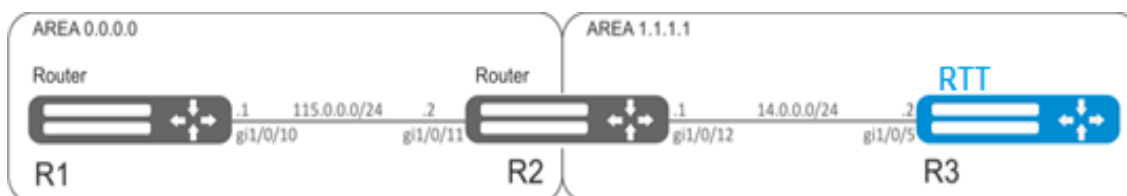


Рисунок 40 – Схема сети

Решение:

Предварительно нужно настроить протокол OSPF и IP-адреса на интерфейсах согласно схеме, приведенной на рисунке 40.

Изменим тип области на тупиковый. На каждом маршрутизаторе из области 1.1.1.1 в режиме конфигурирования области выполним команду:

```
RTT(config-ospf-area) # area-type stub
```

На тупиковом маршрутизаторе R3 включим анонсирование маршрутной информации из протокола RIP:

```
RTT(config-ospf) # redistribute rip
```

7.20.4 Пример настройки Virtual link

Задача:

Объединить две магистральные области в одну с помощью virtual link.

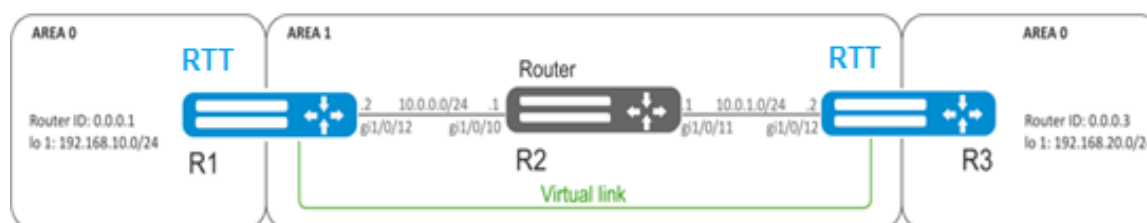


Рисунок 41 – Схема сети

Решение:

Virtual link — это специальное соединение, которое позволяет соединять разорванную на части зону или присоединить зону к магистральной через другую зону. Настраивается между двумя пограничными маршрутизаторами зоны (Area Border Router, ABR).

Предварительно нужно настроить протокол OSPF и IP-адреса на интерфейсах согласно схеме, приведенной на рисунке 41.

На маршрутизаторе R1 перейдем в режим конфигурирования области 1.1.1.1:

```
RTT(config-ospf) # area 1.1.1.1
```

Создадим virtual link с идентификатором 0.0.0.3 и включим его:

```
RTT(config-ospf-area) # virtual-link 0.0.0.3
RTT(config-ospf-vlink) # enable
```

На маршрутизаторе R3 перейдем в режим конфигурирования области 1.1.1.1:

```
RTT(config-ospf) # area 1.1.1.1
```

Создадим virtual link с идентификатором 0.0.0.1 и включим его:

```
RTT(config-ospf-area) # virtual-link 0.0.0.1
RTT(config-ospf-vlink) # enable
```

Рассмотрим таблицу маршрутизации на маршрутизаторе R1:

```
RTT# show ip route
```

C	*	10.0.0.0/24	[0/0]	dev gil/0/12,	[direct 00:49:34]	
O	*	10.0.1.0/24	[150/20]	via 10.0.0.1 on gil/0/12,	[ospf1 00:49:53]	(0.0.0.3)
O	*	192.168.20.0/24	[150/30]	via 10.0.0.1 on gil/0/12,	[ospf1 00:50:15]	(0.0.0.3)
C	*	192.168.10.0/24	[0/0]	dev lo1,	[direct 21:32:01]	

Рассмотрим таблицу маршрутизации на маршрутизаторе R3:

```
RTT# show ip route
```

O	*	10.0.0.0/24	[150/20]	via 10.0.1.1 on gil/0/12,	[ospf1 14:38:35]	(0.0.0.2)
C	*	10.0.1.0/24	[0/0]	dev gil/0/12,	[direct 14:35:34]	
C	*	192.168.20.0/24	[0/0]	dev lo1,	[direct 14:32:58]	
O	*	192.168.10.0/24	[150/30]	via 10.0.1.1 on gil/0/12,	[ospf1 14:39:54]	(0.0.0.1)

Так как OSPF считает виртуальный канал частью области, в таблице маршрутизации R1 маршруты, полученные от R3, отмечены как внутризональные и наоборот.

Для просмотра соседей можно воспользоваться следующей командой:

```
RTT# show ip ospf neighbors 10
```

Таблицу маршрутов протокола OSPF можно просмотреть командой:

```
RTT# show ip ospf 10
```



В firewall необходимо разрешить протокол OSPF (89).

7.21 Настройка BGP

Протокол BGP предназначен для обмена информацией о достижимости подсетей между автономными системами (далее АС), то есть группами маршрутизаторов под единым техническим управлением, использующими протокол внутрисетевой маршрутизации для определения маршрутов внутри себя и протокол междоменной маршрутизации для определения маршрутов доставки пакетов в другие АС. Передаваемая информация включает в себя список АС, к которым имеется доступ через данную систему. Выбор наилучших маршрутов осуществляется исходя из правил, принятых в сети.

7.21.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить приоритетность протокола BGP маршрутизации для основной таблицы маршрутизации (не	<code>RTT(config)# ip protocols bgp preference <VALUE></code>	<VALUE> – приоритетность протокола, принимает значения в диапазоне [1..255]. Значение по умолчанию: BGP (170).
2	Настроить емкость таблиц маршрутизации протокола BGP (не обязательно).	<code>RTT(config)# ip protocols bgp max-routes <VALUE></code> <code>RTT(config)# ipv6 protocols bgp max-routes <VALUE></code> <code>RTT(config-vrf)# ip protocols bgp max-routes <VALUE></code> <code>RTT(config-vrf)# ipv6 protocols bgp max-routes <VALUE></code>	<VALUE> – количество маршрутов протокола BGP в маршрутной таблице, принимает значения в диапазоне: для RTT-R800 [1..2800000]; для RTT-R100/200 [1..1500000]. Значение по умолчанию: BGP (2600000).
3	Включить вывод информации о состоянии отношений с соседями для протокола маршрутизации BGP (не обязательно).	<code>RTT(config)# router bgp log-neighbor-changes</code> <code>RTT(config)# ipv6 router bgp log-neighbor-changes</code>	
4	Включить ESRP и определяется максимальное количество равноценных маршрутов до цели.	<code>RTT(config)# router bgp maximum-paths <VALUE></code>	<VALUE> – количество допустимых равноценных маршрутов до цели, принимает значения [1..16].
5	Создать списки IP-подсетей, которые в дальнейшем будут использоваться для фильтрации анонсируемых и получаемых IP-маршрутов.	<code>RTT(config)# ip prefix-list <NAME></code> <code>RTT(config)# ipv6 prefix-list <NAME></code>	<NAME> – имя конфигурируемого списка подсетей, задаётся строкой до 31 символа.
6	Разрешить (permit) или запретить (deny) списки префиксов.	<code>RTT(config-pl)# permit {object-group <OBJ-GROUP-NETWORK-NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default-route}</code> <code>RTT(config-pl)# deny {object-group <OBJ-GROUP-NETWORK-NAME> [{ eq <LEN> le <LEN> ge <LEN> [le <LEN>] }] default-route}</code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP -адресов, задаётся строкой до 31 символа; <LEN> – длина префикса, принимает значения [1..32] в IP-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; default-route – фильтрация маршрута по умолчанию.
7	Добавить BGP-процесс в систему и осуществить переход в режим настройки параметров BGP-процесса.	<code>RTT(config)# router bgp <AS></code>	<AS> – номер автономной системы процесса, принимает значения [1..4294967295].
8	Определить тип конфигурируемой маршрутной информации и перейти в данный режим настройки.	<code>RTT(config-bgp)# address-family { ipv4 ipv6 } [vrf <VRF>]</code>	– ipv4 – семейство IPv4; – ipv6 – семейство IPv6; <VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать протокол маршрутизации.
9	Установить идентификатор маршрутизатора.	<code>RTT(config-bgp-af)# router-id <ID></code>	<ID> – идентификатор маршрутизатора, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT(config-ipv6-bgp-af)# router-id <ID></code>	
10	Установить временной интервал, по истечении которого идет проверка соединения со встречной стороной.	<code>RTT(config-bgp-af)# timers keepalive <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 60 секунд.
		<code>RTT(config-ipv6-bgp-af)# timers keepalive <TIME></code>	
11	Установить временной интервал, по истечении которого встречная сторона считается недоступной.	<code>RTT(config-bgp-af)# timers holdtime <TIME></code>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 180 секунд.
		<code>RTT(config-ipv6-bgp-af)# timers holdtime <TIME></code>	
12	Установить время минимальной и максимальной задержки, в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения	<code>RTT(config-bgp-af)# timers error-wait <TIME1> <TIME2></code>	<TIME1> – время минимальной задержки в секундах, принимает значения [1..65535]; <TIME2> – время максимальной задержки в секундах, принимает значения [1..65535].
		<code>RTT(config-ipv6-bgp-af)# timers error-wait <TIME1> <TIME2></code>	
13	Установить идентификатор Route-Reflector кластера, которому принадлежит BGP-процесс маршрутизатора.	<code>RTT(config-bgp-af)# cluster-id <ID></code>	<ID> – идентификатор Route-Reflector кластера, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT(config-ipv6-bgp-af)# cluster-id <ID></code>	
14	Определить глобальный алгоритм аутентификации с соседями.	<code>RTT(config-bgp-af)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм шифрования: md5 – пароль шифруется по алгоритму md5.
		<code>RTT(config-ipv6-bgp-af)# authentication algorithm <ALGORITHM></code>	

15	Установить глобальный пароль для аутентификации с соседями.	RTT(config-bgp-af) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<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...).
		RTT(config-ipv6-bgp-af) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	
16	Активировать BGP-процесс.	RTT(config-bgp-af) # enable	
		RTT(config-ipv6-bgp-af) # enable	
17	Включить анонсирование статических маршрутов полученным альтернативным образом.	RTT(config-bgp-af) # redistribute static [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых статических маршрутов, задаётся строкой до 31 символа.
		RTT(config-ipv6-bgp-af) # redistribute static [route-map <NAME>]	
		RTT(config-bgp-af) # redistribute connected [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых напрямую подключённых подсетей, задаётся строкой до 31 символа.
		RTT(config-ipv6-bgp-af) # redistribute connected [route-map <NAME>]	
		RTT(config-bgp-af) # redistribute rip [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых RIP-маршрутов, задаётся строкой до 31 символа.
		RTT(config-ipv6-bgp-af) # redistribute rip [route-map <NAME>]	
		RTT(config-bgp-af) # redistribute ospf <ID> <ROUTE-TYPE> [route-map <NAME>]	<ID> – номер процесса, может принимать значение [1..65535]; <ROUTE-TYPE> – тип маршрута: - 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; <NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых OSPF-маршрутов, задаётся строкой до 31 символа.
		RTT(config-ipv6-bgp-af) # redistribute ospf <ID> <ROUTE-TYPE> [route-map <NAME>]	
		RTT(config-bgp-af) # redistribute bgp <AS> [route-map <NAME>]	<AS> – номер автономной системы, может принимать значения [1..4294967295];

		RTT (config-ipv6-bgp-af) # redistribute bgp <AS> [route-map <NAME>]	<NAME> – имя маршрутной карты, которая будет использоваться для фильтрации и модификации анонсируемых BGP-маршрутов, задаётся строкой до 31 символа.
18	Включить анонсирование подсетей.	RTT (config-bgp-af) # network <ADDR/LEN>	<ADDR/LEN> – адрес подсети, указывается в следующем формате: 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].
		RTT (config-ipv6-bgp-af) # network <ADDR/LEN>	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].
19	Добавить фильтрацию подсетей во входящих или исходящих обновлениях (не обязательно).	RTT (config-bgp-af) # prefix-list <PREFIX-LIST-NAME> { in out }	<PREFIX-LIST-NAME> – имя сконфигурированного списка подсетей, задаётся строкой до 31 символа. in – фильтрация входящих маршрутов; out – фильтрация анонсируемых маршрутов.
20	Добавить BGP-соседа и осуществить переход в режим настройки параметров BGP-соседа.	RTT (config-bgp-af) # neighbor <ADDR>	<ADDR> – IP-адрес соседа, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		RTT (config-ipv6-bgp-af) # neighbor <IPV6-ADDR>	<IPV6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
21	Задать описание соседа. (не обязательно).	RTT (config-bgp-neighbor) # description <DESCRIPTION>	<DESCRIPTION> – описание соседа, задаётся строкой до 255 символов.
		RTT (config-ipv6-bgp-neighbor) # description <DESCRIPTION>	
22	Установить временной интервал, по истечении которого идет проверка соединения со встречной стороной. (не обязательно)	RTT (config-bgp-neighbor) # timers keepalive <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 60 секунд.
		RTT (config-ipv6-bgp-neighbor) # timers keepalive <TIME>	
23	Установить временной интервал, по истечении которого встречная сторона считается недоступной. (не обязательно)	RTT (config-bgp-neighbor) # timers holdtime <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 180 секунд.
		RTT (config-ipv6-bgp-neighbor) # timers holdtime <TIME>	
24	Установить время минимальной и максимальной задержки,	RTT (config-bgp-af) # timers error-wait <TIME1> <TIME2>	<TIME1> – время минимальной задержки в секундах, принимает значения [1..65535];

	в течение которого запрещено устанавливать соединение, в целях защиты от частых разрывов соединения. (не обязательно)	<code>RTT(config-ipv6-bgp-af)# timers error-wait <TIME1> <TIME2></code>	<TIME2> – время максимальной задержки в секундах, принимает значения [1..65535]. Значение по умолчанию: 60 и 300 секунд
25	Установить номер автономной системы BGP-соседа.	<code>RTT(config-bgp-neighbor)# remote-as <AS></code> <code>RTT(config-ipv6-bgp-neighbor)# remote-as <AS></code>	<AS> – номер автономной системы, принимает значения [1..4294967295].
26	Разрешить подключение к соседям, которые находятся не в напрямую подключенных подсетях. (не обязательно)	<code>RTT(config-bgp-neighbor)# ebgp-multihop <NUM></code> <code>RTT(config-ipv6-bgp-neighbor)# ebgp-multihop <NUM></code>	<NUM> - Максимальное количество хопов при установке EBGП (используется для TTL).
27	Задать режим, в котором все обновления отправляются BGP-соседу с указанием в качестве next-hop IP-адреса исходящего интерфейса локального маршрутизатора. (не обязательно)	<code>RTT(config-bgp-neighbor)# next-hop-self</code> <code>RTT(config-ipv6-bgp-neighbor)# next-hop-self</code>	
28	Задать режим, в котором перед отправлением обновления из BGP-атрибута AS Path маршрутов удаляются приватные номера автономных систем (в соответствии с RFC 6996). (не обязательно)	<code>RTT(config-bgp-neighbor)# remove-private-as</code> <code>RTT(config-ipv6-bgp-neighbor)# remove-private-as</code>	
29	Задать режим, в котором BGP-соседу в обновлении наряду с другими маршрутами всегда отправляется маршрут по умолчанию. (не обязательно)	<code>RTT(config-bgp-neighbor)# default-originate</code> <code>RTT(config-ipv6-bgp-neighbor)# default-originate</code>	
30	Включить генерацию и отправку маршрута по умолчанию, если маршрут по умолчанию есть в таблице маршрутизации FIB. (не обязательно)	<code>RTT(config-bgp-af)# default-information-originate</code>	
31	Указать, что BGP-сосед является Route-Reflector	<code>RTT(config-bgp-neighbor)# route-reflector-client</code>	

	клиентом. (не обязательно)	<code>RTT(config-ipv6-bgp-neighbor)# route-reflector-client</code>	
32	Определить приоритетность маршрутов, получаемых от соседа. (не обязательно)	<code>RTT(config-bgp-neighbor)# preference <VALUE></code> <code>RTT(config-ipv6-bgp-neighbor)# preference <VALUE></code>	<VALUE> – приоритетность маршрутов соседа, принимает значения в диапазоне [1..255]. Значение по умолчанию: 170.
33	Задать IP/IPv6-адрес маршрутизатора, который будет использоваться в качестве IP/IPv6-адреса источника в отправляемых обновлениях маршрутной информации BGP. (не обязательно)	<code>RTT(config-bgp-neighbor)# update-source { <ADDR> <IPv6-ADDR> }</code> <code>RTT(config-ipv6-bgp-neighbor)# update-source <ADDR></code>	<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, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
34	Включить режим, в котором разрешен приём маршрутов в BGP-атрибуте, AS Path которых содержит номера автономной системы процесса. (не обязательно)	<code>RTT(config-bgp-neighbor)# allow-local-as <NUMBER></code> <code>RTT(config-bgp-neighbor)# allow-local-as <NUMBER></code>	<NUMBER> – пороговое число вхождений номера автономной системы процесса в атрибуте AS Path, при которых маршрут будет принят, диапазон допустимых значений [1..10].
35	Включить BFD-протокол на конфигурируемом BGP-соседе. (не обязательно)	<code>RTT(config-bgp-neighbor)# bfd-enable</code> <code>RTT(config-ipv6-bgp-neighbor)# bfd-enable</code>	
36	Определить алгоритм аутентификации с соседом. (не обязательно)	<code>RTT(config-bgp-neighbor)# authentication algorithm <ALGORITHM></code> <code>RTT(config-ipv6-bgp-neighbor)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм шифрования: md5 – пароль шифруется по алгоритму md5.
37	Установить пароль для аутентификации с соседом. (не обязательно)	<code>RTT(config-bgp-neighbor)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code> <code>RTT(config-ipv6-bgp-neighbor)# authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<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...).

Часто бывает, особенно при конфигурировании iBGP, что в одном `bgp address-family` необходимо настроить несколько `bgp neighbor` с одинаковыми параметрами. Во избежание избыточности конфигурации рекомендуется использовать `bgp peer-group`, в которой возможно описать общие параметры, а в конфигурации `bgp neighbor` просто указать причастность к `bgp peer-group`.

7.21.2 Пример настройки

Задача:

Настроить BGP-протокол на маршрутизаторе со следующими параметрами:

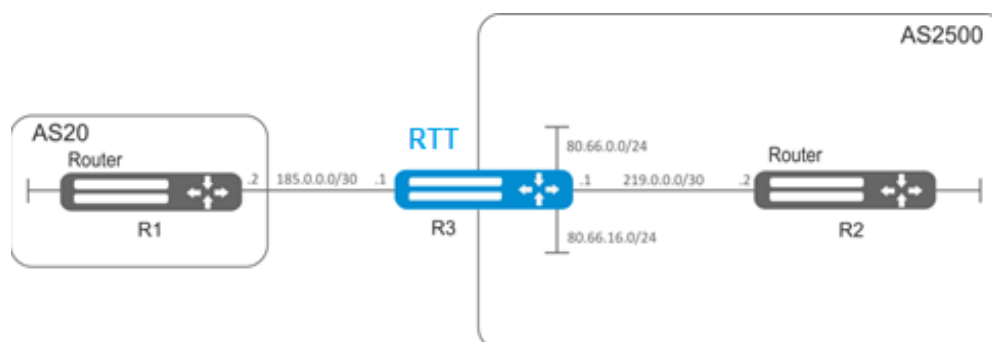


Рисунок 42 – Схема сети

- собственные подсети: 80.66.0.0/24, 80.66.16.0/24;
- анонсирование подсетей, подключенных напрямую;
- собственная AS 2500;
- первое соседство - подсеть 219.0.0.0/30, собственный IP-адрес 219.0.0.1, IP-адрес соседа 219.0.0.2, AS 2500;
- второе соседство - подсеть 185.0.0.0/30, собственный IP-адрес 185.0.0.1, IP-адрес соседа 185.0.0.2, AS 20.

Решение:

Сконфигурируем необходимые сетевые параметры:

```
RTT# configure
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip address 185.0.0.1/30
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# ip address 219.0.0.1/30
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/3
RTT(config-if-gi)# ip address 80.66.0.1/24
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/4
RTT(config-if-gi)# ip address 80.66.16.1/24
RTT(config-if-gi)# exit
```

Создадим BGP процесс для AS 2500 и войдем в режим конфигурирования параметров процесса:

```
RTT(config)# router bgp 2500
```

Входим в режим конфигурирования маршрутной информации для IPv4:

```
RTT(config-bgp)# address-family ipv4
```

Объявим подсети, подключённые напрямую:

```
RTT(config-bgp-af) # redistribute connected
```

Создадим соседства с 185.0.0.2, 219.0.0.2 с указанием автономных систем и включим их:

```
RTT(config-bgp-af) # neighbor 185.0.0.2
RTT(config-bgp-neighbor) # remote-as 20
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
RTT(config-bgp-af) # neighbor 219.0.0.2
RTT(config-bgp-neighbor) # remote-as 2500
RTT(config-bgp-neighbor) # enable
RTT(config-bgp-neighbor) # exit
```

Включим работу протокола:

```
RTT(config-bgp-af) # enable
RTT(config-bgp-af) # exit
RTT(config) # exit
```

Информацию о BGP-пирах можно посмотреть командой:

```
RTT# show ip bgp 2500 neighbors
```

Таблицу маршрутов протокола BGP можно просмотреть с помощью команды:

```
RTT# show ip bgp
```



Необходимо в firewall разрешить TCP-порт 179.

7.22 Настройка BFD

BFD (Bidirectional Forwarding Detection) — это протокол, работающий поверх других протоколов, позволяющий сократить время обнаружения проблемы до 50 мс. BFD является двусторонним протоколом, т.е. требует настройки обоих маршрутизаторов (оба маршрутизатора генерируют BFD-пакеты и отвечают друг-другу).

7.22.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Активировать BFD для протокола OSPF на интерфейсе	RTT(config-if-gi) # ip ospf bfd-enable	
2	Активировать BFD для протокола BGP neighbor на интерфейсе	RTT(config-bgp-neighbor) # bfd-enable	
3	Задать интервал, по истечении которого происходит отправка BFD-сообщения соседу. Глобально (не обязательно)	RTT(config) # ip bfd idle-tx-interval <TIMEOUT>	<TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200 По умолчанию 1 секунда

4	Включить логирование изменений состояния BFD-протокола (не обязательно)	<code>RTT(config)# ip bfd log-adjacency-changes</code>	
5	Задать минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. Глобально (не обязательно)	<code>RTT(config)# ip bfd min-rx-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200 По умолчанию: 300 миллисекунд на RTT-R100/200 200 миллисекунд на RTT-R800
6	Задать минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. Глобально (не обязательно)	<code>RTT(config)# ip bfd min-tx-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200 По умолчанию: 300 миллисекунд на RTT-R100/200 200 миллисекунд на RTT-R800
7	Задать число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. Глобально	<code>RTT(config)# ip bfd multiplier <COUNT></code>	<COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100]. По умолчанию: 5
8	Запустить работу механизма BFD с определенным IP-адресом.	<code>RTT(config)# ip bfd neighbor <ADDR> [{ interface <IF> tunnel <TUN> }] [local-address <ADDR> [multihop]] [vrf <VRF>]</code>	<ADDR> – IP-адрес шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; <IF> – интерфейс или группы интерфейсов; <TUN> – тип и номер туннеля; <VRF> – имя экземпляра VRF, задается строкой до 31 символа; multihop – ключ для установки TTL=255, для работы механизма BFD через маршрутизируемую сеть.
9	Перевести BFD-сессию в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа. Глобально (не обязательно)	<code>RTT(config)# ip bfd passive</code>	

10	Задать интервал, по истечении которого происходит отправка BFD-сообщения соседу. На интерфейсе (не обязательно)	<code>RTT(config-if-gi)# ip bfd idle-tx-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого происходит отправка BFD-пакета, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200. По умолчанию: 1 секунда
11	Задать минимальный интервал, по истечении которого сосед должен сгенерировать BFD-сообщение. На интерфейсе (не обязательно)	<code>RTT(config-if-gi)# ip bfd min-rx-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200 По умолчанию: 300 миллисекунд на RTT-R100/200 200 миллисекунд на RTT-R800
12	Задать минимальный интервал, по истечении которого происходит отправка BFD-сообщения соседу. На интерфейсе (не обязательно)	<code>RTT(config-if-gi)# ip bfd min-tx-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого должна происходить отправка BFD-сообщения соседом, принимает значение в миллисекундах в диапазоне [200..65535] для RTT-R800 и [300..65535] для RTT-R100/200 По умолчанию: 300 миллисекунд на RTT-R100/200 200 миллисекунд на RTT-R800
13	Задать число пропущенных пакетов, после достижения которого BFD-сосед считается недоступным. На интерфейсе (не обязательно)	<code>RTT(config-if-gi)# ip bfd multiplier <COUNT></code>	<COUNT> – число пропущенных пакетов, после достижения которого сосед считается недоступным, принимает значение в диапазоне [1..100]. По умолчанию: 5
14	Перевести BFD-сессию в пассивный режим, то есть BFD-сообщения не будут отправляться до тех пор, пока не будут получены сообщения от BFD-соседа. На интерфейсе (не обязательно)	<code>RTT(config-if-gi)# ip bfd passive</code>	

7.22.2 Пример настройки BFD с BGP

Задача:

Необходимо настроить eBGP между RTT R1 и R2 и включить BFD.



Рисунок 7.43 – Схема сети

Решение:

1. Конфигурирование R1

Предварительно необходимо настроить интерфейс Gi1/0/1:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.0.0.1/24
```

Настроим eBGP с BFD:

```
RTT(config)# router bgp 100
RTT(config-bgp)# address-family ipv4
RTT(config-bgp-af)# neighbor 10.0.0.2
RTT(config-bgp-neighbor)# remote-as 200
RTT(config-bgp-neighbor)# update-source 10.0.0.1
RTT(config-bgp-neighbor)# bfd-enable
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# ex
RTT(config-bgp-af)# enable
RTT(config-bgp-af)# exit
```

2. Конфигурирование R2

Предварительно необходимо настроить интерфейс Gi1/0/1:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 10.0.0.2/24
```

Настроим eBGP с BFD:

```
RTT(config)# router bgp 200
RTT(config-bgp)# address-family ipv4
RTT(config-bgp-af)# neighbor 10.0.0.1
RTT(config-bgp-neighbor)# remote-as 100
RTT(config-bgp-neighbor)# update-source 10.0.0.2
RTT(config-bgp-neighbor)# bfd-enable
RTT(config-bgp-neighbor)# enable
RTT(config-bgp-neighbor)# ex
RTT(config-bgp-af)# enable
RTT(config-bgp-af)# exit
```

7.23 Настройка политики маршрутизации PBR

7.23.1 Настройка Route-map для BGP

Route-map могут служить фильтрами, позволяющими обрабатывать маршрутную информацию при приеме этой информации от соседа либо при ее передаче соседу. Обработка может включать в себя фильтрацию на основании различных признаков маршрута, а также установку атрибутов (MED, AS-PATH, community, LocalPreference и другое) на соответствующие маршруты.

Также Route-map может назначать маршруты на основе списков доступа (ACL).

7.23.1.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать маршрутную карту для фильтрации и модификации IP-маршрутов.	<code>RTT(config)# route-map <NAME></code>	<NAME> – имя маршрутной карты, задается строкой до 31 символа.
2	Создать правило маршрутной карты.	<code>RTT(config-route-map)# rule <ORDER></code>	<ORDER> – номер правила, принимает значения [1 .. 10000].
3	Указать действие, которое должно быть применено для маршрутной информации.	<code>RTT(config-route-map-rule)# action <ACT></code>	<ACT> – назначаемое действие: permit – прием или анонсирование маршрутной информации разрешено; deny – запрещено.
4	Задать значение атрибута BGPAS-Path в маршруте, для которого должно срабатывать правило (не обязательно).	<code>RTT(config-route-map-rule)# match as-path [begin end contain] <AS-PATH></code>	<AS-PATH> – список номеров автономных систем, задается в виде AS,AS,AS, принимает значения [1..4294967295]. Опциональные параметры: begin – значение атрибута начинается с указанных номеров AS; end – значение атрибута заканчивается указанными номерами AS; contain – значение атрибута содержит указанный список номеров AS.
5	Задать значение атрибута BGPCommunity, для которого должно срабатывать правило (не обязательно).	<code>RTT(config-route-map-rule)# match community <COMMUNITY-LIST></code>	<COMMUNITY-LIST> – список community, задается в виде AS:N,AS:N, принимает значения [1..4294967295]. Можно указать до 64 community.
6	Задать значение атрибута BGPExtendedCommunity, для которого должно срабатывать правило (не обязательно).	<code>RTT(config-route-map-rule)# match extcommunity <EXTCOMMUNITY-LIST></code>	<EXTCOMMUNITY-LIST> – список extcommunity, задается в виде KIND:AS:N, KIND:AS:N, где KIND – тип extcommunity: - RT (Route Target); - RO (Route Origin); N – номер extcommunity, принимает значения [1..65535].

7	Задать профиль IP-адресов, содержащий значения подсетей назначения в маршруте (не обязательно).	RTT(config-route-map-rule)# match ip address object-group <OBJ-GROUP- NETWORK - NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задаётся строкой до 31 символа.
		RTT(config-route-map-rule)# match ipv6 address object-group <OBJ-GROUP- NETWORK - NAME>	
8	Задать профиль IP-адресов, содержащий значения атрибута BGPNext-Hop в маршруте для которого должно срабатывать правило (не обязательно).	RTT(config-route-map-rule)# match ip next-hop object-group <OBJ-GROUP- NETWORK - NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.
		RTT(config-route-map-rule)# match ipv6 next-hop object-group <OBJ-GROUP- NETWORK - NAME>	
9	Задать профиль, содержащий IP-адреса маршрутизатора, анонсировавшего маршрут, для которого должно срабатывать правило (не обязательно).	RTT(config-route-map-rule)# match ip route-source object-group <OBJ-GROUP- NETWORK -NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего префиксы подсетей назначения, задается строкой до 31 символа.
		RTT(config-route-map-rule)# match ipv6 route-source object-group <OBJ-GROUP- NETWORK -NAME>	
10	Задать ACL группу, для которой должно срабатывать правило.	RTT(config-route-map-rule)# match access-group <NAME>	<NAME> – имя списка контроля доступа, задается строкой до 31 символа.
11	Задать значение атрибута BGP MED в маршруте для которого должно срабатывать правило (не обязательно).	RTT(config-route-map-rule)# match metric bgp <METRIC>	<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].
12	Задать значение атрибута OSPF Metric в маршруте, для которого должно срабатывать правило.	RTT(config-route-map-rule)# match metric ospf <TYPE> <METRIC>	<TYPE> – тип атрибута OSPF Metric, принимает значение type-1 и type-2; <METRIC> – значение атрибута OSPF Metric, принимает значения [0..65535].
13	Задать значение атрибута RIP Metric в маршруте, для которого должно срабатывать правило.	RTT(config-route-map-rule)# match metric rip <METRIC>	<METRIC> – значение атрибута RIP Metric, принимает значения [0..16].
14	Задать значение атрибута OSPF Tag в маршруте, для которого должно срабатывать правило.	RTT(config-route-map-rule)# match tag ospf <TAG>	<TAG> – значение атрибута OSPF Tag, принимает значения [0..4294967295].
15	Задать значение атрибута RIP Tag в маршруте, для которого должно срабатывать правило.	RTT(config-route-map-rule)# match tag rip <TAG>	<RIP> – значение атрибута RIP Tag, принимает значения [0..65535].

16	Задать значение атрибута BGP AS-Path, которое будет добавляться в начало списка AS-Path (не обязательно).	<code>RTT(config-route-map-rule)# action set as-path prepend <AS-PATH> {track <TRACK-ID>}</code>	<AS-PATH> – список номеров автономных систем, который будет добавлен к текущему значению в маршруте. Задаётся в виде AS,AS,AS, принимает значения [1..4294967295]. <TRACK-ID> – идентификатор vrrp-tracking, при котором будет исполняться указанное действие. Изменяется в диапазоне [1..60].
17	Задать значение атрибута BGP Community, которое будет установлено в маршруте (не обязательно)	<code>RTT(config-route-map-rule)# action set community {COMMUNITY-LIST} no-advertise no-export }</code>	<COMMUNITY-LIST> – список community, задается в виде AS:N,AS:N, где каждая часть принимает значения [1..65535]; no-advertise – маршруты, передаваемые с данным community, не должны анонсироваться другим BGP-соседям; no-export – маршруты, передаваемые с таким community, не должны анонсироваться eBGP-соседям, но анонсируются внешним соседям в конфедерации.
18	Задать значение атрибута BGP ExtCommunity, которое будет установлено в маршруте (не обязательно).	<code>RTT(config-route-map-rule)# action set extcommunity <EXTCOMMUNITY-LIST></code>	<EXTCOMMUNITY-LIST> – список extcommunity, задается в виде KIND:AS:N, KIND:AS:N, где KIND – тип extcommunity: - RT (Route Target); - RO (Route Origin); N – номер extcommunity, принимает значения [1..65535].
19	Задать атрибут BGP Next-Hop, который будет установлен в маршруте при анонсировании (не обязательно).	<code>RTT(config-route-map-rule)# action set ip bgp-next-hop <ADDR></code>	<ADDR> – IP-адрес шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT(config-route-map-rule)# action set ipv6 bgp-next-hop <IPV6-ADDR></code>	<IPV6-ADDR> – IPv6-адрес шлюза, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].

20	Задать значение Next-Hop, которое будет установлено в маршруте, полученном по BGP (не обязательно).	RTT(config-route-map-rule)# action set ip next-hop {NEXTHOP} blackhole unreachable prohibit}	<NEXTHOP> – IP-адрес шлюза задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255]; blackhole – пакеты до данной подсети будут удаляться без отправки уведомлений отправителю; unreachable – пакеты до данной подсети будут удаляться, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1); prohibit – пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMPDestinationunreachable (Communication administratively prohibited code 13).
		RTT(config-route-map-rule)# action set ipv6 next-hop <IPV6-NEXTHOP>	<IPV6-NEXTHOP> – IPv6-адрес шлюза, задается в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
21	Задать значение атрибута BGP Local Preference, который будет установлен в маршруте (не обязательно).	RTT(config-route-map-rule)# action set local-preference <PREFERENCE>	<PREFERENCE> – значение атрибута BGP Local Preference, принимает значения [0..255].
22	Задать значение атрибута BGP Origin, которое будет установлено в маршруте (не обязательно).	RTT(config-route-map-rule)# action set origin <ORIGIN>	<ORIGIN> – значение атрибута BGP Origin: egp – маршрут выучен по протоколу EGP; igp – маршрут получен внутри исходной AS; incomplete – маршрут выучен другим образом.
23	Задать значение BGP MED, которое будет установлено в маршруте (не обязательно).	RTT(config-route-map-rule)# action set metric bgp <METRIC>	<METRIC> – значение атрибута BGP MED, принимает значения [0..4294967295].
24	Добавить фильтрацию и модификацию маршрутов во входящих или исходящих направлениях.	RTT(config-bgp-neighbor)# route-map <NAME><DIRECTION>	<NAME> – имя сконфигурированной маршрутной карты; <DIRECTION> – направление: in – фильтрация и модификация получаемых маршрутов; out – фильтрация и модификация анонсируемых маршрутов.
		RTT(config-ipv6-bgp-neighbor)# route-map <NAME><DIRECTION>	

7.23.1.2 Пример настройки 1

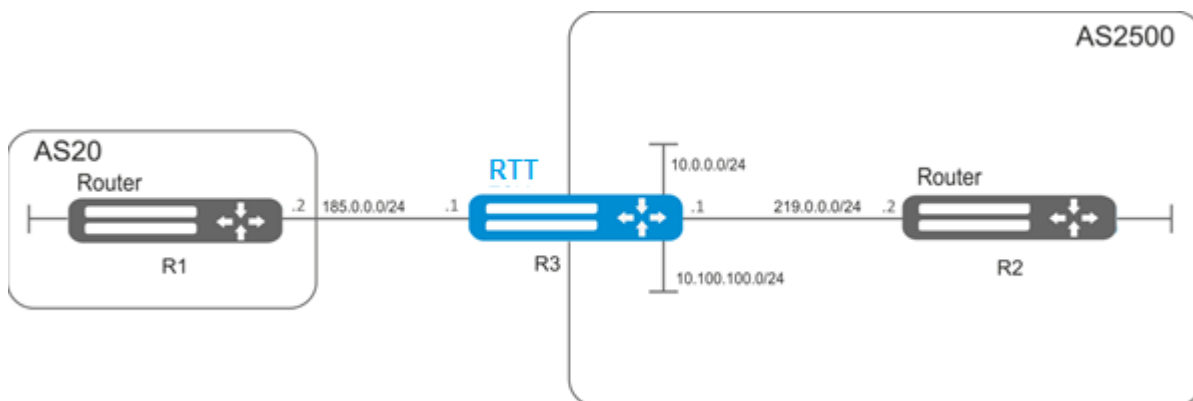


Рисунок 44 – Схема сети

Задача:

Назначить community для маршрутной информации, приходящей из AS 20:

Предварительно нужно выполнить следующие действия:

- Настроить BGP с AS 2500 на маршрутизаторе RTT;
- Установить соседство с AS20.

Решение:

Создаем политику:

```
RTT# configure
RTT(config)# route-map from-as20
```

Создаем правило 1:

```
RTT(config-route-map)# rule 1
```

Если AS PATH содержит AS 20, то назначаем ему community 20:2020 и выходим:

```
RTT(config-route-map-rule)# match as-path contain 20
RTT(config-route-map-rule)# action set community 20:2020
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
```

В BGP процессе AS 2500 заходим в настройки параметров соседа:

```
RTT(config)# router bgp 2500
RTT(config-bgp)# address-family ipv4
RTT(config-bgp-af)# neighbor 185.0.0.2
```

Привязываем политику к принимаемой маршрутной информации:

```
RTT(config-bgp-neighbor)# route-map from-as20 in
```

7.23.1.3 Пример настройки 2

Задача:

Для всей передаваемой маршрутной информации (с community 2500:25) назначить MED, равный 240, и указать источник маршрутной информации EGP:

Предварительно:

Настроить BGP с AS 2500 на RTT

Решение:

Создаем политику:

```
RTT(config)# route-map to-as20
```

Создаем правило:

```
RTT(config-route-map)# rule 1
```

Если community содержит 2500:25, то назначаем ему MED 240 и Origin EGP:

```
RTT(config-route-map-rule)# match community 2500:25
RTT(config-route-map-rule)# action set metric bgp 240
RTT(config-route-map-rule)# action set origin egp
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
```

В BGP процессе AS 2500 заходим в настройки параметров соседа:

```
RTT(config)# router bgp 2500
RTT(config-bgp)# address-family ipv4
RTT(config-bgp-af)# neighbor 185.0.0.2
```

Привязываем политику к анонсируемой маршрутной информации:

```
RTT(config-bgp-neighbor)# route-map to-as20 out
RTT(config-bgp-neighbor)# exit
RTT(config-bgp)# exit
RTT(config)# exit
```

7.23.2 Route-map на основе списков доступа (Policy-based routing)

7.23.2.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать маршрутную карту для фильтрации и модификации IP-маршрутов.	RTT(config)# route-map <NAME>	<NAME> – имя маршрутной карты, задаётся строкой до 31 символа.
2	Создать правило маршрутной карты	RTT(config-route-map)# rule <ORDER>	<ORDER> – номер правила, принимает значения [1 .. 10000].

3	Указать действие, которое должно быть применено для маршрутной информации.	<code>RTT(config-route-map-rule)# action <ACT></code>	<ACT> – назначаемое действие: permit – прием или анонсирование маршрутной информации разрешено; deny – запрещено.
4	Задать ACL, для которого должно срабатывать правило (не обязательно).	<code>RTT(config-route-map-rule)# match ip access-group <NAME></code>	<NAME> – имя списка контроля доступа, задаётся строкой до 31 символа.
5	Задать Next-Хоп для пакетов, которые попадают под критерии в указанном списке доступа (ACL) (не обязательно).	<code>RTT(config-route-map-rule)# action set ip next-hop verify-availability <NEXTHOP><METRIC></code>	<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].
6	Назначить политику маршрутизации на основе списков доступа (ACL).	<code>RTT(config-if-gi)# ip policy route-map <NAME></code>	<NAME> – имя сконфигурированной политики маршрутизации, строка до 31 символа.

7.23.2.2 Пример настройки

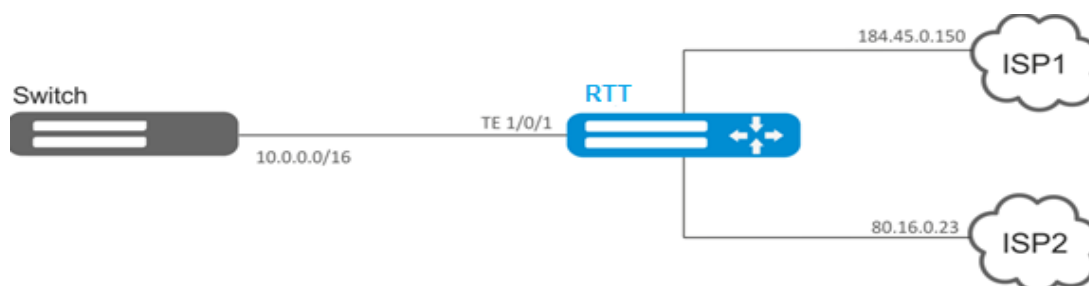


Рисунок 45 – Схема сети

Задача:

Распределить трафик между Интернет провайдерами на основе подсетей пользователей.

Предварительно нужно назначить IP-адреса на интерфейсы.

Требуется направлять трафик с адресов 10.0.20.0/24 через ISP1 (184.45.0.150), а трафик с адресов 10.0.30.0/24 – через ISP2 (80.16.0.23). Требуется контролировать доступность адресов провайдеров (работоспособность подключений к ISP), и при неработоспособности одного из подключений переводить с него на рабочее подключение весь трафик.

Решение:

Создаем ACL:

```

RTT# configure
RTT(config)# ip access-list extended sub20
RTT(config-acl)# rule 1
RTT(config-acl-rule)# match source-address 10.0.20.0 255.255.255.0
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
  
```

```
RTT(config-acl)# exit
RTT(config)# ip access-list extended sub30
RTT(config-acl)# rule 1
RTT(config-acl-rule)# match source-address 10.0.30.0 255.255.255.0
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
```

Создаем политику:

```
RTT(config)# route-map PBR
```

Создаем правило 1:

```
RTT(config-route-map)# rule 1
```

Указываем список доступа (ACL) в качестве фильтра:

```
RTT(config-route-map-rule)# match ip access-group sub20
```

Указываем next-hop для sub20:

```
RTT(config-route-map-rule)# action set ip next-hop verify-availability 184.45.0.150 10
RTT(config-route-map-rule)# action set ip next-hop verify-availability 80.16.0.23 30
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
```

Правилом 1 будет обеспечена маршрутизация трафика из сети 10.0.20.0/24 на адрес 184.45.0.150, а при его недоступности – на адрес 80.16.0.23. Приоритетность шлюзов задается значениями метрик – 10 и 30.

Создаем правило 2:

```
RTT(config-route-map)# rule 2
```

Указываем список доступа (ACL) в качестве фильтра:

```
RTT(config-route-map-rule)# match ip access-group sub30
```

Указываем nexthop для sub30 и выходим:

```
RTT(config-route-map-rule)# action set ip next-hop verify-availability 80.16.0.23 10
RTT(config-route-map-rule)# action set ip next-hop verify-availability 184.45.0.150 30
RTT(config-route-map-rule)# exit
RTT(config-route-map)# exit
```

Правилом 2 будет обеспечена маршрутизация трафика из сети 10.0.30.0/24 на адрес 80.16.0.23, а при его недоступности – на адрес 184.45.0.150. Приоритетность задается значениями метрик.

Заходим на интерфейс TE 1/0/1:

```
RTT(config)# interface tengigabitethernet 1/0/1
```

Привязываем политику на соответствующий интерфейс:

```
RTT(config-if-te) # ip policy route-map PBR
```

7.24 Настройка GRE-туннелей

GRE (англ. Generic Routing Encapsulation — общая инкапсуляция маршрутов) — протокол туннелирования сетевых пакетов. Его основное назначение — инкапсуляция пакетов сетевого уровня сетевой модели OSI в IP-пакеты. GRE может использоваться для организации VPN на 3-м уровне модели OSI. В маршрутизаторе RTT реализованы статические неуправляемые GRE-туннели, то есть туннели создаются вручную путем конфигурирования на локальном и удаленном узлах. Параметры туннеля для каждой из сторон должны быть взаимосогласованными или переносимые данные не будут декапсулироваться партнером.

7.24.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать L3-интерфейс, от которого будет строиться GRE-туннель.		
2	Создать GRE-туннель и перейти в режим его конфигурирования.	<code>RTT(config)# tunnel gre <INDEX></code>	<INDEX> – идентификатор туннеля в диапазоне: для RTT-R100/200 – [1..250], для RTT-R800 - [1..500]
3	Указать экземпляр VRF, в котором будет работать данный GRE-туннель (не обязательно).	<code>RTT(config-bridge)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Указать описание конфигурируемого туннеля (не обязательно).	<code>RTT(config-gre)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
5	Установить локальный IP-адрес для установки туннеля.	<code>RTT(config-gre)# local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT(config-gre)# interface <IF></code>	<IF> – интерфейс, от IP-адреса которого устанавливается туннель.
6	Установить удаленный IP-адрес для установки туннеля.	<code>RTT(config-gre)# remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
7	Указать режим инкапсуляции для GRE туннеля.	<code>RTT(config-gre)# mode <MODE></code>	<MODE> - режим инкапсуляции для GRE туннеля: ip – инкапсуляция IP-пакетов в GRE; ethernet – инкапсуляция Ethernet-фреймов в GRE. Значение по умолчанию: ip
8	Установить IP-адрес локальной стороны туннеля (только в режиме ip).	<code>RTT(config-gre)# ip address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и префикс подсети задается в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает

			значения [1..32]. Можно указать до 8 IP-адресов перечислением через запятую.
9	Назначить широковещательный домен для инкапсуляции в GRE-пакеты данного туннеля (только в режиме ethernet).	RTT(config-gre) # bridge-group <BRIDGE-ID>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: для RTT-R100/200 – [1..250], для RTT-R800 – [1..500]
10	Указать размер MTU (MaximumTransmissionUnit) для туннеля (не обязательно). MTU более 1500 будет активно только если применена команда "system jumbo-frames"	RTT(config-gre) # mtu <MTU>	<MTU> – значение MTU, принимает значения в диапазоне: [1280..10000]. Значение по умолчанию: 1500.
11	Указать значение времени жизни TTL для туннельных пакетов (не обязательно).	RTT(config-gre) # ttl <TTL>	<TTL> – значение TTL, принимает значения в диапазоне [1..255]. Значение по умолчанию: Наследуется от инкапсулируемого пакета.
12	Указать DSCP для использования в IP-заголовке инкапсулирующего пакета (не обязательно).	RTT(config-gre) # dscp <DSCP>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: наследуется от инкапсулируемого пакета.
13	Разрешить передачу ключа (Key) в туннельном заголовке GRE (в соответствии с RFC 2890) и установить значение ключа. Настраивается с обеих сторон туннеля. (не обязательно).	RTT(config-gre) # key <KEY>	<KEY> – значение KEY, принимает значения в диапазоне [1..2000000]. Значение по умолчанию: ключ не передаётся.
14	Включить вычисление контрольной суммы и занесение её в GRE-заголовок отправляемых пакетов. При этом на удаленной стороне необходимо включить проверку контрольной суммы. (не обязательно)	RTT(config-gre) # local checksum	
15	Включить проверку наличия и соответствия значений контрольной суммы в заголовках принимаемых GRE-пакетов. При этом на удаленной стороне необходимо включить вычисление контрольной суммы. (не обязательно)	RTT(config-gre) # remote checksum	
16	Включить проверку доступности удаленного шлюза туннеля (не обязательно)	RTT(config-gre) # keepalive enable	

17	Задать время ожидания keepalive пакетов от встречной стороны (не обязательно)	<code>RTT(config-gre) # keepalive timeout <TIME></code>	<TIME> – время в секундах, принимает значения в диапазоне [1..32767]. Значение по умолчанию: 10
18	Задать количество попыток проверки доступности удаленного шлюза туннеля (не обязательно)	<code>RTT(config-gre) # keepalive retries <VALUE></code>	<VALUE> – количество попыток, принимает значения в диапазоне [1..255]. Значение по умолчанию: 5
19	Задать интервал времени, за который усредняется статистика о нагрузке на туннеле (не обязательно)	<code>RTT(config-gre) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5
20	Включить отправку snmp-trap о включении/отключении туннеля.	<code>RTT(config-gre) # snmp init-trap</code>	
21	Включить механизм перезапроса IP-адресов по протоколу DHCP на указанных интерфейсах при отключении GRE-туннеля по keepalive (не обязательно)	<code>RTT(config-gre) # keepalive dhcp dependent-interface <IF></code>	<IF> – физический/логический интерфейс, на котором включено получение IP-адреса по DHCP
22	Задать интервал времени между отключением GRE-туннеля и перезапросом IP-адреса на интерфейсе/интерфейсах, указанных командой keepalive dhcp dependent-interface (не обязательно)	<code>RTT(config-gre) # keepalive dhcp link- timeout <SEC></code>	<SEC> – интервал между отключением GRE-туннеля и перезапросом IP-адреса по DHCP на интерфейсах
23	Активировать туннель.	<code>RTT(config-gre) # enable</code>	

7.24.2 Пример настройки IP-GRE-туннеля

Задача:

Организовать L3-VPN между офисами компании через IP-сеть, используя для туннелирования трафика протокол GRE.

- в качестве локального шлюза для туннеля используется IP-адрес 115.0.0.1;
- в качестве удаленного шлюза для туннеля используется IP-адрес 114.0.0.10;
- IP-адрес туннеля на локальной стороне 25.0.0.1/24.



Рисунок 46 – Схема сети

Решение:

Предварительно на маршрутизаторах должны быть настроены интерфейсы для связи с сетью WAN разрешено получение пакетов протокола GRE из зоны безопасности, в которой работают интерфейсы, подключенные к сети WAN.

Создадим туннель GRE 10:

```
RTT(config)# tunnel gre 10
```

Укажем локальный и удаленный шлюз (IP-адреса интерфейсов, граничащих с WAN):

```
RTT(config-gre)# local address 115.0.0.1  
RTT(config-gre)# remote address 114.0.0.10
```

Укажем IP-адрес туннеля 25.0.0.1/24:

```
RTT(config-gre)# ip address 25.0.0.1/24
```

Также туннель должен принадлежать к зоне безопасности, для того чтобы можно было создать правила, разрешающие прохождение трафика в firewall. Принадлежность туннеля к зоне задается следующей командой:

```
RTT(config-gre)# security-zone untrusted
```

Включим туннель:

```
RTT(config-gre)# enable  
RTT(config-gre)# exit
```

На маршрутизаторе должен быть создан маршрут до локальной сети партнера. В качестве интерфейса назначения указываем ранее созданный туннель GRE:

```
RTT(config)# ip route 172.16.0.0/16 tunnel gre 10
```

После применения настроек трафик будет инкапсулироваться в туннель и отправляться партнеру, независимо от наличия GRE-туннеля и правильности настроек с его стороны.

Опционально для GRE-туннеля можно указать следующие параметры:

- Включить вычисление и включение в пакет контрольной суммы заголовка GRE и инкапсулированного пакета для исходящего трафика:

```
RTT(config-gre)# local checksum
```
- Включить проверку наличия и корректности контрольной суммы GRE для входящего трафика:

```
RTT(config-gre)# remote checksum
```
- Указать уникальный идентификатор:

```
RTT(config-gre)# key 15808
```
- Указать значение DSCP, MTU, TTL:

```
RTT(config-gre)# dscp 44  
RTT(config-gre)# mtu 1426  
RTT(config-gre)# ttl 18
```

- Включить и настроить механизм keepalive:

```
RTT(config-gre)# keepalive enable
RTT(config-gre)# keepalive timeout <TIME>
RTT(config-gre)# keepalive retries <VALUE>
```

Состояние туннеля можно посмотреть командой:

```
RTT# show tunnels status gre 10
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
RTT# show tunnels counters gre 10
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show tunnels configuration gre 10
```

Настройка туннеля IPv4-over-IPv4 производится аналогичным образом.



При создании туннеля необходимо в firewall разрешить протокол GRE(47).

7.25 Настройка L2TPv3-туннелей

L2TPv3 (Layer 2 Tunneling Protocol Version 3) – протокол для туннелирования пакетов 2-го уровня модели OSI между двумя IP-узлами. В качестве инкапсулирующего протокола используется IP или UDP. L2TPv3 может использоваться как альтернатива MPLS P2P L2VPN (VLL) для организации VPN уровня L2. В маршрутизаторе RTT реализованы статические неуправляемые L2TPv3-туннели, то есть туннели создаются вручную путем конфигурирования на локальном и удаленном узлах. Параметры туннеля на каждой из сторон должны быть взаимосогласованными или переносимые данные не будут декапсулироваться партнером.

7.25.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать L3-интерфейс, от которого будет строиться L2TPv3-туннель.		
2	Создать L2TPv3-туннель и перейти в режим его конфигурирования.	RTT(config)# tunnel l2tpv3 <INDEX>	<INDEX> – идентификатор туннеля в диапазоне: для RTT-R100/200 – [1..250], для RTT-R800 - [1..500]
3	Указать описание конфигурируемого туннеля (не обязательно).	RTT(config-l2tpv3)# description <DESCRIPTION>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
4	Указать экземпляр VRF, в котором будет работать данный L2TPv3-туннель (не обязательно).	RTT(config-l2tpv3)# ip vrf forwarding <VRF>	<VRF> – имя VRF, задается строкой до 31 символа.
5	Установить локальный IP-адрес для установки туннеля.	RTT(config-l2tpv3)# local address <ADDR>	<ADDR> – IP-адрес локального шлюза, задаётся в виде

			AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
6	Установить удаленный IP-адрес для установки туннеля.	RTT(config-l2tpv3) # remote address <ADDR>	<ADDR> – IP-адрес локального шлюза, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
7	Выбрать метод инкапсуляции для туннеля L2TPv3.	RTT(config-l2tpv3) # protocol <TYPE>	<TYPE> – тип инкапсуляции, возможные значения: ip -инкапсуляция в IP-пакет; udp -инкапсуляция в UDP-дейтаграммы.
8	Установить локальный идентификатор сессии.	RTT(config-l2tpv3) # local session-id <SESSION-ID>	<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].
9	Установить удаленный идентификатор сессии.	RTT(config-l2tpv3) # remote session-id <SESSION-ID>	<SESSION-ID> – идентификатор сессии, принимает значения [1..200000].
10	Определить локальный UDP-порт (если в качестве метода инкапсуляции был выбран UDP протокол).	RTT(config-l2tpv3) # local port <UDP>	<UDP> – номер UDP-порта в диапазоне [1..65535].
11	Определить удаленный UDP-порт (если в качестве метода инкапсуляции был выбран UDP протокол).	RTT(config-l2tpv3) # remote port <UDP>	<UDP> – номер UDP-порта в диапазоне [1..65535].
12	Назначить широковещательный домен для инкапсуляции в L2TPV3-пакеты данного туннеля.	RTT(config-l2tpv3) # bridge-group <BRIDGE-ID>	<BRIDGE-ID> – идентификационный номер моста, принимает значения в диапазоне: RTT-R100/200 – [1..250], для RTT-R800 - [1..500]
13	Активировать туннель.	RTT(config-l2tpv3) # enable	
14	Указать размер MTU (MaximumTransmissionUnit) для туннелей (не обязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames"	RTT(config-l2tpv3) # mtu <MTU>	<MTU> – значение MTU, принимает значения в диапазоне: [1280..10000]. Значение по умолчанию: 1500.
15	Определить локальное значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией (не обязательно).	RTT(config-l2tpv3) # local cookie <COOKIE>	<COOKIE> – значение COOKIE, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде.
16	Определить удаленное значение cookie для дополнительной проверки соответствия между передаваемыми данными и сессией (не обязательно).	RTT(config-l2tpv3) # remote cookie <COOKIE>	<COOKIE> – значение COOKIE, параметр принимает значения длиной восемь или шестнадцать символов в шестнадцатеричном виде.
17	Задать интервал времени, за который усредняется статистика о нагрузке на туннеле (не обязательно)	RTT(config-l2tpv3) # load-average <TIME>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5.

7.25.2 Пример настройки L2TPv3-туннеля

Задача:

Организовать L2 VPN между офисами компании через IP-сеть, используя для туннелирования трафика протокол L2TPv3.

- в качестве инкапсулирующего протокола используется UDP, номер порта на локальной стороне и номер порта на стороне партнера 519;
- в качестве локального шлюза для туннеля используется IP-адрес 21.0.0.1;
- в качестве удаленного шлюза для туннеля используется IP-адрес 183.0.0.10;
- идентификатор туннеля на локальной стороне равен 2, на стороне партнера 3;
- идентификатор сессии внутри туннеля равен 100, на стороне партнера 200;
- в туннель направим трафик из bridge с идентификатором 333.



Рисунок 47 – Схема сети

Решение:

Создадим туннель L2TPv3 333:

```
RTT# configure
```

```
RTT(config)# tunnel l2tpv3 333
```

Укажем локальный и удаленный шлюз (IP-адреса интерфейсов, граничащих с WAN):

```
RTT(config-l2tpv3)# local address 21.0.0.1
RTT(config-l2tpv3)# remote address 183.0.0.10
```

Укажем тип инкапсулирующего протокола и номера UDP-портов:

```
RTT(config-l2tpv3)# protocol udp
RTT(config-l2tpv3)# local port 519
RTT(config-l2tpv3)# remote port 519
```

Укажем идентификаторы сессии внутри туннеля для локальной и удаленной сторон:

```
RTT(config-l2tpv3)# local session-id 100
RTT(config-l2tpv3)# remote session-id 200
```

Установим принадлежность L2TPv3-туннеля к мосту, который должен быть связан с сетью удаленного офиса (настройка моста рассматривается в пункте 0):

```
RTT(config-l2tpv3)# bridge-group 333
```

Включим ранее созданный туннель и выйдем:

```
RTT(config-l2tpv3)# enable
```

```
RTT(config-l2tpv3) # exit
```

Создадим саб-интерфейс для коммутации трафика, поступающего из туннеля, в локальную сеть с тегом VLAN id 333:

```
RTT(config) # interface gi 1/0/2.333
```

Установим принадлежность саб-интерфейса к мосту, который должен быть связан с локальной сетью (настройка моста рассматривается в пункте 7.17):

```
RTT(config-subif) # bridge-group 333
```

```
RTT(config-subif) # exit
```

После применения настроек трафик будет инкапсулироваться в туннель и отправляться партнеру, независимо от наличия L2TPv3 туннеля и правильности настроек с его стороны.

Настройки туннеля в удаленном офисе должны быть зеркальными локальным. В качестве локального шлюза должен использоваться IP-адрес 183.0.0.10. В качестве удаленного шлюза должен использоваться IP-адрес 21.0.0.1. Номер порта инкапсулирующего протокола на локальной стороне и стороне партнера 519. Идентификатор сессии внутри туннеля должен быть равным 200, на стороне партнера 100. Также туннель должен принадлежать мосту, который необходимо соединить с сетью партнера.

Состояние туннеля можно посмотреть командой:

```
RTT# show tunnels status l2tpv3 333
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
RTT# show tunnels counters l2tpv3 333
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show tunnels configuration l2tpv3 333
```



Помимо создания туннеля необходимо в firewall разрешить входящий трафик по протоколу UDP с портом отправителя 519 и портом назначения 519.

7.26 Настройка IPsec VPN

IPsec – это набор протоколов, которые обеспечивают защиту передаваемых с помощью IP-протокола данных. Данный набор протоколов позволяет осуществлять подтверждение подлинности (аутентификацию), проверку целостности и шифрование IP-пакетов, а также включает в себя протоколы для защищённого обмена ключами в сети Интернет.

7.26.1 Настройка Route-based IPsec VPN

7.26.1.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать VTI-туннель и перейти в	RTT(config) # tunnel vti <TUN>	<TUN> – имя туннеля устройства.

	режим его конфигурирования.		
2	Указать локальный IP-адрес VTI-туннеля.	<code>RTT(config-vti)#local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза.
3	Указать удаленный IP-адрес VTI-туннеля.	<code>RTT(config-vti)#remote address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.
4	Установить IP-адрес локальной стороны VTI-туннеля	<code>RTT(config-vti)# ip address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и префикс подсети задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].
5	Включить VTI-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall для VTI-туннеля.	<code>RTT(config-vti)# security-zone<NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.
		<code>RTT(config-vti)# ip firewall disable</code>	
6	Включить туннель.	<code>RTT(config-vti)#enable</code>	
7	Создать IKE-профиль и перейти в режим его конфигурирования.	<code>RTT(config)# security ike proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
8	Указать описание конфигурируемого IKE-профиля (не обязательно).	<code>RTT(config-ike-proposal)# description<DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
9	Определить алгоритм аутентификации для IKE. (не обязательно)	<code>RTT(config-ike-proposal)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512. Значение по умолчанию: sha1
10	Определить алгоритм шифрования для IKE. (не обязательно)	<code>RTT(config-ike-proposal)# encryption algorithm <ALGORITHM></code>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256. Значение по умолчанию: 3des
10	Определить номер группы Диффи-Хеллмана. (не обязательно)	<code>RTT(config-ike-proposal)# dh-group <DH-GROUP></code>	<DH-GROUP> – номер группы Диффи-Хеллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18]. Значение по умолчанию: 1

11	Определить режим аутентификации IKE. (не обязательно)	<code>RTT(config-ike-proposal) # authentication method <METHOD></code>	<METHOD> - метод аутентификации ключа. Может принимать значения: pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования; rsa-public-key – метод аутентификации, использующий RSA-сертификат. Значение по умолчанию: pre-shared-key
12	Создать ike-политику и перейти в режим её конфигурирования.	<code>RTT(config) # security ike policy <NAME></code>	<NAME> – имя политики IKE, задаётся строкой до 31 символа.
13	Задать время жизни соединения протокола IKE (не обязательно).	<code>RTT(config-ike-proposal) # lifetime seconds <SEC></code>	<SEC> – период времени, принимает значения [4 ..86400] секунд. Значение по умолчанию: 3600
14	Привязать IKE-профиль к IKE-политике.	<code>RTT(config-ike-policy) # proposal <NAME></code>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
15	Указать ключ аутентификации. (обязательно, если в качестве режима аутентификации выбран pre-shared-key)	<code>RTT(config-ike-policy) # pre-shared-key ascii-text<TEXT></code>	<TEXT> – строка [1..64] ASCII символов.
16	Создать IKE-шлюз и перейти в режим его конфигурирования.	<code>RTT(config) # security ike gateway <NAME></code>	<NAME> – имя шлюза протокола IKE, задаётся строкой до 31 символа.
17	Привязать IKE-политику к IKE-шлюзу.	<code>RTT(config-ike-gw) # ike-policy <NAME></code>	<NAME> – имя политики протокола IKE, задаётся строкой до 31 символа.
18	Указать версию IKE (не обязательно).	<code>RTT(config-ike-gw) # version <VERSION></code>	<version> – версия IKE-протокола: v1-only или v2-only. Значение по умолчанию: v1-only
19	Установить режим перенаправления трафика в туннель - route-based.	<code>RTT(config-ike-gw) # mode - route-based</code>	
20	Указать действие для DPD (не обязательно).	<code>RTT(config-ike-gw) # dead-peer-detection action <MODE></code>	<MODE> – режим работы DPD: restart – соединение переустанавливается; clear – соединение останавливается; hold – соединение поддерживается; none – механизм выключен, никаких действий не предпринимается. Значение по умолчанию: none

21	Указать интервал между отправкой сообщений механизмом DPD (не обязательно).	RTT(config-ike-gw) # dead-peer-detection interval <SEC>	<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд. Значение по умолчанию: 2
22	Указать период времени для ответа на сообщения механизма DPD (не обязательно).	RTT(config-ike-gw) # dead-peer-detection timeout <SEC>	<SEC> – период времени для ответа на сообщения механизма DPD, принимает значения [1..180] секунд. Значение по умолчанию: 30 секунд
23	Привязать VTI-туннель к IKE-шлюзу.	RTT(config-ike-gw) # bind-interface vti <VTI>	<VTI> – идентификационный номер интерфейса VTI.
24	Создать в IPsec-профиль.	RTT(config) # security ipsec proposal <NAME>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
25	Определить алгоритм аутентификации для IPsec. (не обязательно)	RTT(config-ipsec- proposal) # authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512. Значение по умолчанию: sha1
26	Определить алгоритм шифрования для IPsec. (не обязательно)	RTT(config-ipsec- proposal) # encryption algorithm <ALGORITHM>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256. Значение по умолчанию: 3des
27	Указать протокол инкапсуляции для IPsec (не обязательно).	RTT(config-ipsec- proposal) # protocol <PROTOCOL>	<PROTOCOL> – инкапсулирующий протокол, принимает значения: esp, ah Значение по умолчанию: esp
28	Создать IPsec-политику и перейти в режим её конфигурирования.	RTT(config) # security ipsec policy <NAME>	<NAME> – имя политики IPsec, задаётся строкой до 31 символа.
29	Привязать IPsec-профиль к IPsec-политике	RTT(config-ipsec- policy) # proposal <NAME>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.

30	Задать время жизни IPsec-туннеля (не обязательно).	<code>RTT(config-ipsec-policy)# lifetime { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – период времени жизни IPsec-туннеля, по истечении происходит пересогласование. Принимает значения [1140..86400] секунд.</p> <p><PACKETS> – количество пакетов, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400].</p> <p><KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400] секунд. Значение по умолчанию: 28800 секунд</p>
31	Создать IPsec VPN и перейти в режим конфигурирования.	<code>RTT(config)# security ipsecvpn <NAME></code>	<NAME> – имя VPN, задаётся строкой до 31 символа.
32	Определить режим согласования данных, необходимых для активации VPN.	<code>RTT(config-ipsec-vpn)# mode <MODE></code>	<MODE> – режим работы VPN.
33	Привязать IPsec-политику к IPsec-VPN.	<code>RTT(config-ipsec-vpn)# ike ipsec-policy <NAME></code>	<NAME> – имя IPsec-политики, задаётся строка до 31 символа.
34	Задать значение DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола (не обязательно).	<code>RTT(config-ipsec-vpn)# ike dscp <DSCP></code>	<p>DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63].</p> <p>Значение по умолчанию: 63</p>
34	Установить режим активации VPN.	<code>RTT(config-ipsec-vpn)# ike establish-tunnel <MODE></code>	<p><MODE> – режим активации VPN:</p> <p>by-request – соединение активируется встречной стороной;</p> <p>route – соединение активируется при появлении трафика, маршрутизируемого в туннель;</p> <p>immediate – туннель активируется автоматически после применения конфигурации.</p>
36	Осуществить привязку IKE-шлюза к IPsec-VPN.	<code>RTT(config-ipsec-vpn)# ike gateway <NAME></code>	<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.
37	Установить значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA (не обязательно).	<code>RTT(config-ipsec-vpn)# ike idle-time <TIME></code>	<TIME> – интервал в секундах, принимает значения [4..86400].

38	Отключить пересогласование ключей до разрыва IKE-соединения по истечению времени, количеству переданных пакетов или байт (не обязательно).	<code>RTT(config-ipsec-vpn)# ike rekey disable</code>	
39	Настроить начало пересогласования ключей IKE-соединения до истечения времени жизни (не обязательно).	<code>RTT(config-ipsec-vpn)# ike rekey margin { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<p><SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой <code>lifetimeseconds</code>, см. 22.2.13). Принимает значения [4..86400].</p> <p><PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой <code>lifetimepackets</code>). Принимает значения [4..86400]</p> <p><KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой <code>lifetimekilobytes</code>). Принимает значения [4..86400]</p> <p>Значение по умолчанию: - Пересогласование ключей до истечения времени – за 540 секунд. - Пересогласование ключей до истечения объема трафика и количества пакетов – отключено.</p>
40	Установить уровень случайного разброса значений параметров <code>margin seconds</code> , <code>margin packets</code> , <code>margin kilobytes</code> (не обязательно).	<code>RTT(config-ipsec-vpn)# ike rekey randomization <VALUE></code>	<p><VALUE> – максимальный процент разброса значений, принимает значения [1..100].</p> <p>Значение по умолчанию: 100%</p>
41	Указать описание для IPsec-VPN (не обязательно).	<code>RTT(config-ipsec-vpn)# description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
42	Активировать IPsec VPN .	<code>RTT(config-ipsec-vpn)# enable</code>	

7.26.1.2 Пример настройки



Рисунок 48 – Схема сети

Задача:

Настроить IPsec-туннель между R1 и R2.

- R1 IP-адрес - 120.11.5.1;
- R2 IP-адрес - 180.100.0.1;

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

IPsec:

- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

Решение:

1. Конфигурирование R1

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
RTT# configure
RTT(config)# interface gi 1/0/1
RTT(config-if-gi)# ip address 180.100.0.1/24
RTT(config-if-gi)# security-zone untrusted
RTT(config-if-gi)# exit
```

Создадим туннель VTI. Трафик будет перенаправляться через VTI в IPsec-туннель. В качестве локального и удаленного шлюза указываются IP-адреса интерфейсов, граничащих с WAN:

```
RTT(config)# tunnel vti 1
RTT(config-vti)# local address 180.100.0.1
RTT(config-vti)# remote address 120.11.5.1
RTT(config-vti)# enable
RTT(config-vti)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
RTT(config)# object-group service ISAKMP
RTT(config-object-group-service)# port-range 500
RTT(config-object-group-service)# exit
```

Создадим статический маршрут до удаленной LAN-сети. Для каждой подсети, которая находится за IPsec-туннелем, нужно указать маршрут через VTI-туннель:

```
RTT(config)# ip route 192.0.2.0/24 tunnel vti 1
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

```
RTT(config)# security ike proposal ike_prop1
RTT(config-ike-proposal)# dh-group 2
RTT(config-ike-proposal)# authentication algorithm md5
RTT(config-ike-proposal)# encryption algorithm aes128
RTT(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

```
RTT(config)# security ike policy ike_poll
RTT(config-ike-policy)# pre-shared-key hexadecimal 123FFF
RTT(config-ike-policy)# proposal ike_prop1
RTT(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель:

```
RTT(config)# security ike gateway ike_gw1
RTT(config-ike-gw)# ike-policy ike_poll
RTT(config-ike-gw)# mode route-based
RTT(config-ike-gw)# bind-interface vti 1
RTT(config-ike-gw)# version v2-only
RTT(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
RTT(config)# security ipsec proposal ipsec_prop1
RTT(config-ipsec-proposal)# authentication algorithm md5
RTT(config-ipsec-proposal)# encryption algorithm aes128
RTT(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

```
RTT(config)# security ipsec policy ipsec_poll
RTT(config-ipsec-policy)# proposal ipsec_prop1
RTT(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой *enable*.

```
RTT(config)# security ipsec vpn ipsec1
RTT(config-ipsec-vpn)# mode ike
RTT(config-ipsec-vpn)# ike establish-tunnel immediate
RTT(config-ipsec-vpn)# ike gateway ike_gw1
RTT(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
RTT(config-ipsec-vpn)# enable
RTT(config-ipsec-vpn)# exit
RTT(config)# exit
```

2. Конфигурирование R2

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
RTT# configure
RTT(config)# interface gi 1/0/1
RTT(config-if)# ip address 120.11.5.1/24
RTT(config-if)# security-zone untrusted
RTT(config-if)# exit
```

Создадим туннель VTI. Трафик будет перенаправляться через VTI в IPsec-туннель. В качестве локального и удаленного шлюза указываются IP-адреса интерфейсов, граничащих с WAN:

```
RTT(config)# tunnel vti 1
RTT(config-vti)# remote address 180.100.0.1
```

```
RTT(config-vti)# local address 120.11.5.1
RTT(config-vti)# enable
RTT(config-vti)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
RTT(config)# object-group service ISAKMP
RTT(config-object-group-service)# port-range 500
RTT(config-object-group-service)# exit
```

Создадим статический маршрут до удаленной LAN-сети. Для каждой подсети, которая находится за IPsec-туннелем, нужно указать маршрут через VTI-туннель:

```
RTT(config)# ip route 10.0.0.0/16 tunnel vti 1
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хеллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

```
RTT(config)# security ike proposal ike_prop1
RTT(config-ike-proposal)# dh-group 2
RTT(config-ike-proposal)# authentication algorithm md5
RTT(config-ike-proposal)# encryption algorithm aes128
RTT(config-ike-proposal)# exit
RTT(config)#
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

```
RTT(config)# security ike policy ike_pol1
RTT(config-ike-policy)# pre-shared-key hexadecimal 123FFF
RTT(config-ike-policy)# proposal ike_prop1
RTT(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель:

```
RTT(config)# security ike gateway ike_gw1
RTT(config-ike-gw)# ike-policy ike_pol1
RTT(config-ike-gw)# mode route-based
RTT(config-ike-gw)# bind-interface vti 1
RTT(config-ike-gw)# version v2-only
RTT(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
RTT(config)# security ipsec proposal ipsec_prop1
RTT(config-ipsec-proposal)# authentication algorithm md5
RTT(config-ipsec-proposal)# encryption algorithm aes128
RTT(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

```
RTT(config)# security ipsec policy ipsec_pol1
RTT(config-ipsec-policy)# proposal ipsec_prop1
RTT(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой *enable*.

```
RTT(config)# security ipsec vpn ipsec1
RTT(config-ipsec-vpn)# mode ike
RTT(config-ipsec-vpn)# ike establish-tunnel immediate
RTT(config-ipsec-vpn)# ike gateway ike_gw1
RTT(config-ipsec-vpn)# ike ipsec-policy ipsec_pol1
RTT(config-ipsec-vpn)# enable
RTT(config-ipsec-vpn)# exit
RTT(config)# exit
```

Состояние туннеля можно посмотреть командой:

```
RTT# show security ipsec vpn status ipsec1
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show security ipsec vpn configuration ipsec1
```



В firewall необходимо разрешить протокол ESP и ISAKMP (UDP-порт 500).

7.26.2 Настройка Policy-based IPsec VPN

7.26.2.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать IKE-экземпляр и перейти в режим его конфигурирования.	RTT(config)# security ike proposal <NAME>	<NAME> – имя профиля протокола IKE, задаётся строкой до 31 символа.
2	Указать описание конфигурируемого туннеля (не обязательно).	RTT(config-ike- proposal)# description<DESCRIPTION>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
3	Определить алгоритм аутентификации для IKE.	RTT(config-ike- proposal)# authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.
4	Определить алгоритм шифрования для IKE.	RTT(config-ike- proposal)# encryption algorithm <ALGORITHM>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
5	Определить номер группы	RTT(config-ike-	<DH-GROUP> – номер группы

	Диффи-Хеллмана.	<code>proposal) # dh-group <DH-GROUP></code>	Диффи-Хеллмана, принимает значения [1, 2, 5, 14, 15, 16, 17, 18].
6	Определить режим аутентификации.	<code>RTT(config-ike-proposal) # authentication method <METHOD></code>	<code><METHOD></code> - метод аутентификации ключа. Может принимать значения: pre-shared-key – метод аутентификации, использующий предварительно полученные ключи шифрования; rsa-public-key – метод аутентификации, использующий RSA-сертификат.
7	Создать политику для профиля IKE и перейти в режим её конфигурирования.	<code>RTT(config) # security ike policy <NAME></code>	<code><NAME></code> – имя политики IKE, задаётся строкой до 31 символа.
8	Задать время жизни соединения протокола IKE (не обязательно).	<code>RTT(config-ike-proposal) # lifetime seconds <SEC></code>	<code><SEC></code> – период времени, принимает значения [4 ..86400] секунд.
9	Привязать политику к профилю.	<code>RTT(config-ike-policy) # proposal <NAME></code>	<code><NAME></code> – имя профиля протокола IKE, задаётся строкой до 31 символа.
10	Указать ключ аутентификации.	<code>RTT(config-ike-policy) # pre-shared-key ascii-text<TEXT></code>	<code><TEXT></code> – строка [1..64] ASCII символов.
11	Создать шлюз для IKE и перейти в режим его конфигурирования.	<code>RTT(config) # security ike gateway <NAME></code>	<code><NAME></code> – имя шлюза протокола IKE, задаётся строкой до 31 символа.
12	Привязать политику IKE.	<code>RTT(config-ike-gw) # ike-policy <NAME></code>	<code><NAME></code> – имя политики протокола IKE, задаётся строкой до 31 символа.
13	Указать версию IKE (не обязательно).	<code>RTT(config-ike-gw) # version <VERSION></code>	<code><version></code> – версия IKE-протокола: v1-only или v2-only .
14	Установить режим перенаправления трафика в туннель.	<code>RTT(config-ike-gw) # mode<MODE></code>	<code><MODE></code> – 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.
15	Указать действие для DPD (не обязательно).	<code>RTT(config-ike-gw) # dead-peer-detection action <MODE></code>	<code><MODE></code> – режим работы DPD: restart – соединение переустанавливается; clear – соединение останавливается; hold – соединение поддерживается; none – механизм выключен,

			никаких действий не предпринимается.
16	Указать интервал между отправкой сообщений механизмом DPD (не обязательно).	<code>RTT(config-ike-gw)#dead-peer-detection interval <SEC></code>	<SEC> – интервал между отправкой сообщений механизмом DPD, принимает значения [1..180] секунд.
17	Указать период времени для ответа на сообщения механизма DPD (не обязательно).	<code>RTT(config-ike-gw)#dead-peer-detection timeout <SEC></code>	<SEC> – период времени для ответа на сообщения механизма DPD, принимает значения [1..180] секунд.
18	Указать версию IKE (не обязательно).	<code>RTT(config-ike-gw)#version <VERSION></code>	<version> – версия IKE-протокола: v1-only или v2-only .
19	Установить IP подсети отправителя.	<code>RTT(config-ike-gw)#local network <ADDR/LEN> [protocol { <TYPE> <ID> } [port <PORT>]]</code>	<ADDR/LEN> – IP-адрес и маска подсети 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]; <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].
20	Установить IP-адрес локального шлюза IPsec-туннеля.	<code>RTT(config-ike-gw)#local address <ADDR></code>	<ADDR> – IP-адрес локального шлюза.
21	Установить IP-адрес удаленного шлюза IPsec-туннеля.	<code>RTT(config-ike-gw)#remote address <ADDR></code>	<ADDR> – IP-адрес удаленного шлюза.
22	Установить IP-адрес подсети получателя, а также IP-протокол и порт.	<code>RTT(config-ike-gw)#remote network <ADDR/LEN> [protocol { <TYPE> <ID> } [port <PORT>]]</code>	<ADDR/LEN> – IP-адрес и маска подсети 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]; <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].

23	Создать в профиль IPsec.	<code>RTT(config)# security ipsec proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
24	Определить алгоритм аутентификации для IPsec.	<code>RTT(config-ipsec- proposal)# authentication algorithm <ALGORITHM></code>	<ALGORITHM> – алгоритм аутентификации, принимает значения: md5, sha1, sha2-256, sha2-384, sha2-512.
26	Определить алгоритм шифрования для IPsec.	<code>RTT(config-ipsec- proposal)# encryption algorithm <ALGORITHM></code>	<ALGORITHM> – протокол шифрования, принимает значения: des, 3des, blowfish128, blowfish192, blowfish256, aes128, aes192, aes256, aes128ctr, aes192ctr, aes256ctr, camellia128, camellia192, camellia256.
26	Указать протокол (не обязательно).	<code>RTT(config-ipsec- proposal)#protocol <PROTOCOL></code>	<PROTOCOL> – инкапсулирующий протокол, принимает значения: esp, ah.
27	Создать политику для профиля IPsec и перейти в режим её конфигурирования	<code>RTT(config)# security ipsec policy <NAME></code>	<NAME> – имя политики IPsec, задаётся строкой до 31 символа.
28	Привяжем политику к профилю	<code>RTT(config-ipsec- policy)# proposal <NAME></code>	<NAME> – имя профиля протокола IPsec, задаётся строкой до 31 символа.
29	Задать время жизни IPsec туннеля (не обязательно).	<code>RTT(config-ipsec- policy)# lifetime { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<SEC> – период времени жизни IPsec-туннеля, по истечении которого происходит пересогласование. Принимает значения [1140..86400] секунд. <PACKETS> – количество пакетов, после передачи которых происходит пересогласование IPsec-туннеля. Принимает значения [4..86400]. <KB> – объем трафика, после передачи которого происходит пересогласование IPsec-туннеля. Принимает значения [4..86400] секунд.
30	Создать IPsec VPN и перейти в режим конфигурирования.	<code>RTT(config)# security ipsecvpn<NAME></code>	<NAME> – имя VPN, задаётся строкой до 31 символа.
31	Определить режим согласования данных, необходимых для активации VPN.	<code>RTT(config-ipsec- vpn)# mode <MODE></code>	<MODE> – режим работы VPN.
32	Привязать IPsec политику к VPN.	<code>RTT(config-ipsec- vpn)#ike ipsec- policy <NAME></code>	<NAME> – имя IPsec-политики, задаётся строка до 31 символа.
33	Задать значение DSCP для использования в IP-заголовке исходящих пакетов IKE-протокола (не обязательно).	<code>RTT(config-ipsec- vpn)#ike dscp <DSCP></code>	DSCP – значение кода DSCP, принимает значения в диапазоне [0..63].

34	Устанавливается режим активации VPN.	<code>RTT(config-ipsec-vpn)#ike establish-tunnel <MODE></code>	<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.
35	Осуществить привязка IKE-шлюза к VPN.	<code>RTT(config-ipsec-vpn)#ike gateway <NAME></code>	<NAME> – имя IKE-шлюза, задаётся строкой до 31 символа.
36	Установить значение временного интервала в секундах, по истечению которого соединение закрывается, если не было принято или передано ни одного пакета через SA (не обязательно).	<code>RTT(config-ipsec-vpn)#ike idle-time <TIME></code>	<TIME> – интервал в секундах, принимает значения [4..86400].
37	Отключить пересогласование ключей до разрыва IKE соединения по истечению времени, количеству переданных пакетов или байт (не обязательно).	<code>RTT(config-ipsec-vpn)#ike rekey disable</code>	
38	Настроить начало пересогласования ключей IKE соединения до истечения времени жизни (не обязательно).	<code>RTT(config-ipsec-vpn)#ike rekey margin { seconds <SEC> packets <PACKETS> kilobytes <KB> }</code>	<SEC> – интервал времени в секундах, оставшийся до закрытия соединения (задается командой <code>lifetimeseconds</code>). Принимает значения [4..86400]. <PACKETS> – количество пакетов, оставшихся до закрытия соединения (задается командой <code>lifetimepackets</code>). Принимает значения [4..86400]. <KB> – объем трафика в килобайтах, оставшийся до закрытия соединения (задается командой <code>lifetimekilobytes</code>). Принимает значения [4..86400]
39	Установить уровень случайного разброса значений параметров <code>marginseconds</code> , <code>marginpackets</code> , <code>marginkilobytes</code> (не обязательно).	<code>RTT(config-ipsec-vpn)#ike rekey randomization <VALUE></code>	<VALUE> – максимальный процент разброса значений, принимает значения [1..100].
40	Описать VPN (не обязательно).	<code>RTT(config-ipsec-vpn)#description <DESCRIPTION></code>	<DESCRIPTION> – описание профиля, задаётся строкой до 255 символов.
41	Активировать IPsec VPN.	<code>RTT(config-ipsec-vpn)#enable</code>	

7.26.2.2 Пример настройки

Задача:



Рисунок 49 – Схема сети

Настроить IPsec-туннель между R1 и R2.

R1 IP-адрес - 120.11.5.1;

R2 IP-адрес - 180.100.0.1;

IKE:

- группа Диффи-Хэллмана: 2;
- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

IPsec:

- алгоритм шифрования: AES 128 bit;
- алгоритм аутентификации: MD5.

Решение:

1. Конфигурирование R1

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
RTT# configure
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# ip address 120.11.5.1/24
RTT(config-if-gi)# security-zone untrusted
RTT(config-if-gi)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
RTT(config)# object-group service ISAKMP
RTT(config-object-group-service)# port-range 500
RTT(config-object-group-service)# exit
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

```
RTT(config)# security ike proposal ike_prop1
RTT(config-ike-proposal)# dh-group 2
RTT(config-ike-proposal)# authentication algorithm md5
RTT(config-ike-proposal)# encryption algorithm aes128
RTT(config-ike-proposal)# exit
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, по которым могут согласовываться узлы и ключ аутентификации:

```
RTT(config)# security ike policy ike_poll
RTT(config-ike-policy)# pre-shared-key hexadecimal 123FFF
RTT(config-ike-policy)# proposal ike_prop1
RTT(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель:

```
RTT(config)# security ike gateway ike_gw1
RTT(config-ike-gw)# ike-policy ike_poll
RTT(config-ike-gw)# local address 180.100.0.1
RTT(config-ike-gw)# local network 10.0.0.0/16
RTT(config-ike-gw)# remote address 120.11.5.1
RTT(config-ike-gw)# remote network 192.0.2.0/24
RTT(config-ike-gw)# mode policy-based
RTT(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
RTT(config)# security ipsec proposal ipsec_prop1
RTT(config-ipsec-proposal)# authentication algorithm md5
RTT(config-ipsec-proposal)# encryption algorithm aes128
RTT(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

```
RTT(config)# security ipsec policy ipsec_poll
RTT(config-ipsec-policy)# proposal ipsec_prop1
RTT(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой *enable*.

```
RTT(config)# security ipsec vpn ipsec1
RTT(config-ipsec-vpn)# mode ike
RTT(config-ipsec-vpn)# ike establish-tunnel immediate
RTT(config-ipsec-vpn)# ike gateway ike_gw1
RTT(config-ipsec-vpn)# ike ipsec-policy ipsec_poll
RTT(config-ipsec-vpn)# enable
RTT(config-ipsec-vpn)# exit
RTT(config)# exit
```

2. Конфигурирование R2

Настроим внешний сетевой интерфейс и определим принадлежность к зоне безопасности:

```
RTT# configure
RTT(config)# interface gi 1/0/1
RTT(config-if)# ip address 120.11.5.1/24
RTT(config-if)# security-zone untrusted
RTT(config-if)# exit
```

Для настройки правил зон безопасности потребуется создать профиль порта протокола ISAKMP:

```
RTT(config)# object-group service ISAKMP
RTT(config-addr-set)# port-range 500
RTT(config-addr-set)# exit
```

Создадим профиль протокола IKE. В профиле укажем группу Диффи-Хэллмана 2, алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IKE-соединения:

```
RTT(config)# security ike proposal ike_prop1
RTT(config-ike-proposal)# dh-group 2
RTT(config-ike-proposal)# authentication algorithm md5
RTT(config-ike-proposal)# encryption algorithm aes128
RTT(config-ike-proposal)# exit
RTT(config)#
```

Создадим политику протокола IKE. В политике указывается список профилей протокола IKE, которым могут согласовываться узлы и ключ аутентификации:

```
RTT(config)# security ike policy ike_pol1
RTT(config-ike-policy)# pre-shared-key hexadecimal 123FFF
RTT(config-ike-policy)# proposal ike_prop1
RTT(config-ike-policy)# exit
```

Создадим шлюз протокола IKE. В данном профиле указывается VTI-туннель, политика, версия протокола и режим перенаправления трафика в туннель:

```
RTT(config)# security ike gateway ike_gw1
RTT(config-ike-gw)# ike-policy ike_pol1
RTT(config-ike-gw)# remote address 180.100.0.1
RTT(config-ike-gw)# remote network 10.0.0.0/16
RTT(config-ike-gw)# local address 120.11.5.1
RTT(config-ike-gw)# local network 192.0.2.0/24
RTT(config-ike-gw)# mode policy-based
RTT(config-ike-gw)# exit
```

Создадим профиль параметров безопасности для IPsec-туннеля. В профиле укажем алгоритм шифрования AES 128 bit, алгоритм аутентификации MD5. Данные параметры безопасности используются для защиты IPsec-туннеля:

```
RTT(config)# security ipsec proposal ipsec_prop1
RTT(config-ipsec-proposal)# authentication algorithm md5
RTT(config-ipsec-proposal)# encryption algorithm aes128
RTT(config-ipsec-proposal)# exit
```

Создадим политику для IPsec-туннеля. В политике указывается список профилей IPsec-туннеля, по которым могут согласовываться узлы.

```
RTT(config)# security ipsec policy ipsec_pol1
RTT(config-ipsec-policy)# proposal ipsec_prop1
RTT(config-ipsec-policy)# exit
```

Создадим IPsec VPN. В VPN указывается шлюз IKE-протокола, политика IPsec-туннеля, режим обмена ключами и способ установления соединения. После ввода всех параметров включим туннель командой *enable*.

```
RTT(config)# security ipsec vpn ipsec1
RTT(config-ipsec-vpn)# mode ike
```



```
RTT(config-ipsec-vpn) # ike establish-tunnel immediate
RTT(config-ipsec-vpn) # ike gateway ike_gw1
RTT(config-ipsec-vpn) # ike ipsec-policy ipsec_poll
RTT(config-ipsec-vpn) # enable
RTT(config-ipsec-vpn) # exit
RTT(config) # exit
```

Состояние туннеля можно посмотреть командой:

```
RTT# show security ipsec vpn status ipsec1
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show security ipsec vpn configuration ipsec1
```



В firewall необходимо разрешить протокол ESP и ISAKMP (UDP-порт 500).

7.27 Настройка LT-туннелей

LT (англ. Logical Tunnel - логический туннель) - тип туннелей, предназначенный для передачи маршрутной информации и трафика между различными виртуальными маршрутизаторами (VRF Lite), сконфигурированными на одном аппаратном маршрутизаторе. LT-туннель может использоваться для организации взаимодействия между двумя или более VRF с применением ограничений firewall.

7.27.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать LT-туннели для каждого из существующих VRF.	<code>RTT(config)# tunnel lt <ID></code>	<ID> – идентификатор туннеля в диапазоне [1..128].
2	Указать описание конфигурируемых туннелей (не обязательно).	<code>RTT(config-lt)# description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
3	Включить каждый LT-туннель в соответствующий VRF.	<code>RTT(config-lt)# ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Включить каждый LT-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall для LT-туннеля.	<code>RTT(config-lt)# security-zone<NAME></code> <code>RTT(config-lt)# ip firewall disable</code>	<NAME> – имя зоны безопасности, задаётся строкой до 12 символов.
5	Для каждого LT-туннеля задать номер противоположный LT туннель (в другом VRF).	<code>RTT(config-lt)# peer lt <ID></code>	<ID> – идентификатор туннеля в диапазоне [1..128].
6	Для каждого LT-туннеля указать IP-адрес для маршрутизации пакетов. Для взаимодействующих LT-туннелей, IP-адреса должны быть из одной IP-подсети.	<code>RTT(config-lt)# ip address <ADDR/LEN></code>	<ADDR/LEN> – IP-адрес и префикс подсети, задаётся в виде AAA.BBB.CCC.DDD/EE, где каждая часть AAA – DDD принимает значения [0..255] и EE принимает значения [1..32].

7	Включить туннели.	<code>RTT(config-lt) #enable</code>	
8	Для каждого VRF настроить необходимые протоколы маршрутизации через LT-туннель.		
9	Задать интервал времени, за который усредняется статистика о нагрузке на туннеле (не обязательно)	<code>RTT(config-lt) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150]. Значение по умолчанию: 5
10	Указать размер MTU (Maximum Transmission Unit) пакетов, которые может пропускать данный bridge (не обязательно; возможно, если в bridge включен только VLAN). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames"	<code>RTT(config-lt) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: [552..10000].

7.27.2 Пример настройки

Задача: Организовать взаимодействие между хостами, терминированными в двух VRF vrf_1 и vrf_2.

Исходная конфигурация:

```
hostname RTT
ip vrf vrf_1
exit
ip vrf vrf_2
exit

interface gigabitethernet 1/0/1
 ip vrf forwarding vrf_1
 ip firewall disable
 ip address 10.0.0.1/24
exit
interface gigabitethernet 1/0/2
 ip vrf forwarding vrf_2
 ip firewall disable
 ip address 10.0.1.1/24
exit
```

Решение:

Создадим LT-туннели для каждого VRF с указанием IP-адресов из одной подсети:

```
RTT(config)# tunnel lt 1
RTT(config-lt)# ip vrf forwarding vrf_1
RTT(config-lt)# ip firewall disable
RTT(config-lt)# ip address 192.168.0.1/30
RTT(config-lt)# exit
RTT(config)# tunnel lt 2
RTT(config-lt)# ip vrf forwarding vrf_2
```

```
RTT(config-lt) # ip firewall disable
RTT(config-lt) # ip address 192.168.0.2/30
RTT(config-lt) # exit
```

Укажем для каждого LT-туннеля LT-туннель из VRF, с которым необходимо установить связь, и активируем их:

```
RTT(config) # tunnel lt 1
RTT(config-lt) # peer lt 2
RTT(config-lt) # enable
RTT(config-lt) # exit
RTT(config) # tunnel lt 2
RTT(config-lt) # peer lt 1
RTT(config-lt) # enable
RTT(config-lt) # exit
```



Если в VRF не работает ни один из протоколов динамической маршрутизации, то необходимо указать статические маршруты для каждого VRF:

```
RTT(config) # ip route vrf vrf_1 0.0.0.0/0 192.168.0.2
RTT(config) # ip route vrf vrf_2 0.0.0.0/0 192.168.0.1
```

7.28 Настройка удаленного доступа к корпоративной сети по PPTP-протоколу

PPTP (англ. Point-to-Point Tunneling Protocol) – туннельный протокол типа точка-точка, позволяющий компьютеру устанавливать защищённое соединение с сервером за счёт создания специального туннеля в обычной незащищенной сети. PPTP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. PPTP может также использоваться для организации туннеля между двумя локальными сетями. PPTP использует дополнительное TCP-соединение для обслуживания туннеля.

7.28.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль PPTP-сервера.	RTT(config) # remote-access pptp <NAME>	<NAME> – имя профиля PPTP-сервера, задаётся строкой до 31 символа.
2	Выбрать режим аутентификации PPTP-клиентов.	RTT(config-pptp-server) # authentication mode { local radius }	local – аутентификация пользователя по локальной базе. radius - аутентификация пользователя по базе RADIUS-сервера.
3	Указать описание конфигурируемого сервера (не обязательно).	RTT(config-pptp-server) # description <DESCRIPTION>	<DESCRIPTION> – описание PPTP-сервера, задаётся строкой до 255 символов.
4	Указать список DNS-серверов, которые будут использовать удаленные пользователи (не обязательно).	RTT(config-pptp-server) # dns-servers object-group <OBJ-GROUP-NETWORK-NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.
5	Указать DSCP-приоритет исходящих пакетов (не обязательно).	RTT(config-pptp-server) # dscp <DSCP>	<DSCP> – dscp приоритет исходящих пакетов [0..63].

6	Включить шифрование MPPE для PPTP-соединений (не обязательно).	<code>RTT(config-pptp-server)# encryption mppe</code>	
7	Указать IP-адрес локального шлюза.	<code>RTT(config-pptp-server)# local-address object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит IP-адрес локального шлюза, задаётся строкой до 31 символа; <ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
8	Указать размер MTU (MaximumTransmissionUnit) для сервера (не обязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames"	<code>RTT(config-pptp-server) mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне [1280..1500]. Значение по умолчанию: 1500.
9	Указать IP-адрес, который должен обрабатывать PPTP-сервер.	<code>RTT(config-pptp-server)# outside-address { object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR> }</code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля содержащего IP-адрес, который должен слушать PPTP-сервер задаётся строкой до 31 символа; <ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
10	Указать список IP-адресов, из которого PPTP выдаются динамические IP-адреса удаленным пользователям.	<code>RTT(config-pptp-server)# remote-address { object-group <OBJ-GROUP-NETWORK-NAME> address-range <FROM-ADDR>-<TO-ADDR> }</code>	<OBJ-GROUP-NETWORK-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].
11	Включить PPTP-сервер в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall.	<code>RTT(config-pptp-server)# security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
12	Указать имя пользователя (при использовании локальной аутентификации пользователей).	<code>RTT(config-pptp-server) username <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 12 символов.
13	Указать пароль пользователя.	<code>RTT(config-pptp-user) password ascii-text { <PASSWORD> encrypted <PASSWORD> }</code>	<PASSWORD> - пароль пользователя, задается строкой до 32 символов.
14	Активировать пользователя.	<code>RTT(config-pptp-user) enable</code>	
17	Указать список WINS-серверов, которые будут использовать удаленные пользователи (не обязательно).	<code>RTT(config-pptp-server)# wins-servers object-group <OBJ-GROUP-NETWORK-NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

7.28.2 Пример настройки PPTP-сервера

Задача:

Настроить PPTP-сервер на маршрутизаторе.

- адрес PPTP-сервера – 120.11.5.1;
- шлюз внутри туннеля для подключающихся клиентов – 10.10.10.1;
- пул IP-адресов для выдачи 10.10.10.5-10.10.10.25;
- DNS-серверы: 8.8.8.8, 8.8.8.4;
- учетные записи для подключения – fedor, ivan.

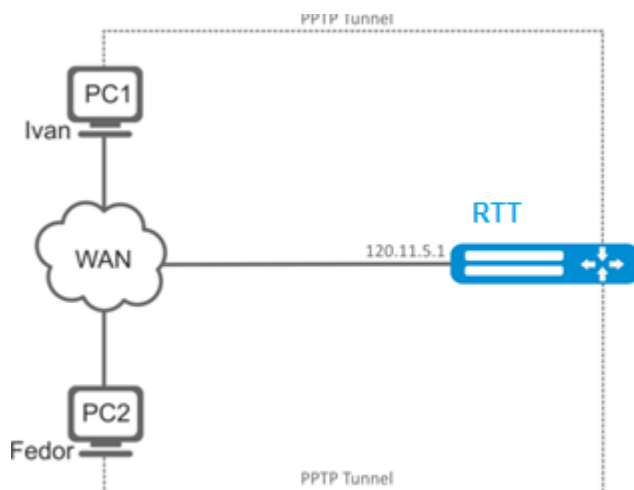


Рисунок 50 – Схема сети

Решение:

Создадим профиль адресов, содержащий адрес, который должен слушать сервер:

```
RTT# configure
RTT(config)# object-group network pptp_outside
RTT(config-object-group-network)# ip address-range 120.11.5.1
RTT(config-object-group-network)# exit
```

Создадим профиль адресов, содержащий адрес локального шлюза:

```
RTT(config)# object-group network pptp_local
RTT(config-object-group-network)# ip address-range 10.10.10.1
RTT(config-object-group-network)# exit
```

Создадим профиль адресов, содержащий адреса клиентов:

```
RTT(config)# object-group network pptp_remote
RTT(config-object-group-network)# ip address-range 10.10.10.5-10.10.10.25
RTT(config-object-group-network)# exit
```

Создадим профиль адресов, содержащий DNS-серверы:

```
RTT(config)# object-group network pptp_dns
RTT(config-object-group-network)# ip address-range 8.8.8.8
RTT(config-object-group-network)# ip address-range 8.8.8.4
RTT(config-object-group-network)# exit
```

Создадим PPTP-сервер и привяжем вышеуказанные профили:

```
RTT(config)# remote-access pptp remote-workers
RTT(config-pptp)# local-address object-group pptp_local
RTT(config-pptp)# remote-address object-group pptp_remote
RTT(config-pptp)# outside-address object-group pptp_outside
RTT(config-pptp)# dns-servers object-group pptp_dns
```

Выберем метод аутентификации пользователей PPTP-сервера:

```
RTT(config-pptp)# authentication mode local
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
RTT(config-pptp)# security-zone VPN
```

Создадим PPTP-пользователей *Ivan* и *Fedor* для PPTP-сервера:

```
RTT(config-pptp)# username ivan
RTT(config-pptp-user)# password ascii-text password1
RTT(config-pptp-user)# enable
RTT(config-pptp-user)# exit
RTT(config-pptp)# username fedor
RTT(config-pptp-user)# password ascii-text password2
RTT(config-pptp-user)# enable
RTT(config-pptp-user)# exit
RTT(config-pptp)# exit
```

Включим PPTP-сервер:

```
RTT(config-pptp)# enable
```

После применения конфигурации маршрутизатор будет прослушивать 120.11.5.1:1723. Состояние сессий PPTP-сервера можно посмотреть командой:

```
RTT# show remote-access status pptp server remote-workers
```

Счетчики сессий PPTP-сервера можно посмотреть командой:

```
RTT# show remote-access counters pptp server remote-workers
```

Очистить счетчики сессий PPTP-сервера можно командой:

```
RTT# clear remote-access counters pptp server remote-workers
```

Завершить сессию пользователя *fedor* PPTP-сервера можно одной из следующих команд:

```
RTT# clear remote-access session pptp username fedor
RTT# clear remote-access session pptp server remote-workers username fedor
```

Конфигурацию PPTP-сервера можно посмотреть командой:

```
RTT# show remote-access configuration pptp remote-workers
```



Помимо создания PPTP-сервера необходимо в firewall открыть TCP-порт 1723 для обслуживания соединений и разрешить протокол GRE(47) для туннельного трафика.

7.29 Настройка удаленного доступа к корпоративной сети по L2TP over IPsec протоколу

L2TP (англ. Layer 2 Tunneling Protocol – протокол туннелирования второго уровня) – туннельный протокол, использующийся для поддержки виртуальных частных сетей. L2TP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. L2TP может также использоваться для организации туннеля между двумя локальными сетями. L2TP использует дополнительное UDP-соединение для обслуживания туннеля. L2TP-протокол не предоставляет средств шифрования данных и поэтому он обычно используется в связке с группой протоколов IPsec, которая предоставляет безопасность на пакетном уровне.

7.29.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль L2TP-сервера.	<code>RTT(config)# remote-access l2tp <NAME></code>	<NAME> – имя профиля L2TP-сервера, задаётся строкой до 31 символа.
2	Выбрать режим аутентификации L2TP-клиентов.	<code>RTT(config-l2tp-server)# authentication mode { local radius }</code>	local – аутентификация пользователя по локальной базе. radius – аутентификация пользователя по базе RADIUS-сервера.
3	Указать описание конфигурируемого сервера (не обязательно).	<code>RTT(config-l2tp-server)# description <DESCRIPTION></code>	<DESCRIPTION> – описание L2TP-сервера, задаётся строкой до 255 символов.
4	Указать список DNS-серверов, которые будут использовать удаленные пользователи (не обязательно).	<code>RTT(config-l2tp-server)# dns-servers object-group <OBJ-GROUP-NETWORK-NAME></code>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, содержащего, который содержит адреса необходимых DNS-серверов, задаётся строкой до 31 символа.
5	Указать DSCP приоритет исходящих пакетов.	<code>RTT(config-l2tp-server)# dscp <DSCP></code>	<DSCP> – dscp приоритет исходящих пакетов [0..63].
6	Включить сервер.	<code>RTT(config-l2tp-server)# enable</code>	
7	Выбрать метод аутентификации по ключу для IKE-соединения.	<code>RTT(config-l2tp-server)# ipsec authentication method pre-shared-key</code>	
8	Указать общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.	<code>RTT(config-l2tp-server)# ipsec authentication pre-shared-key { ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> } hexadecimal { <HEX> encrypted <ENCRYPTED-HEX> } }</code>	<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 [1..32] байт, задаётся строкой [2..128] символов.

			[2..64] байт, задаётся строкой [2..256] символов.
8	Указать IP-адрес локального шлюза	<pre>RTT(config-l2tp-server)# local-address object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR></pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит IP-адрес локального шлюза, задаётся строкой до 31 символа;</p> <p><ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
9	Указать размер MTU (MaximumTransmissionUnit) для сервера (не обязательно). MTU более 1500 будет активно только в случае применения команды "system jumbo-frames"	<pre>RTT(config-l2tp-server) mtu <MTU></pre>	<p><MTU> – значение MTU, принимает значения в диапазоне [1280..1500].</p> <p>Значение по умолчанию: 1500.</p>
10	Указать IP-адрес, который должен слушать L2TP-сервер.	<pre>RTT(config-l2tp-server)# outside-address object-group <OBJ-GROUP-NETWORK-NAME> ip-address <ADDR></pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля содержащего IP-адрес, который должен слушать L2TP-сервер, задаётся строкой до 31 символа;</p> <p><ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
11	Указать список IP-адресов из которого L2TP выдаются динамические IP-адреса удаленным пользователям.	<pre>RTT(config-l2tp-server)# remote-address { object-group <OBJ-GROUP-NETWORK-NAME> address-range <FROM-ADDR>-<TO-ADDR> }</pre>	<p><OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит список IP-адресов удаленных пользователей, задаётся строкой до 31 символа;</p> <p><FROM-ADDR> – начальный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p><TO-ADDR> – конечный IP-адрес диапазона, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p>
12	Включить L2TP-сервер в зону безопасности и настроить правила взаимодействия между зонами.	<pre>RTT(config-l2tp-server)# security-zone <NAME></pre>	<p><NAME> – имя зоны безопасности, задаётся строкой до 31 символа.</p>
13	Указать имя пользователя (при использовании локальной базы аутентификации).	<pre>RTT(config-l2tp-server) username <NAME></pre>	<p><NAME> – имя пользователя, задаётся строкой до 12 символов.</p>
14	Указать пароль пользователя (при использовании локальной базы аутентификации).	<pre>RTT(config-l2tp-user) password ascii-text { <PASSWORD> encrypted <PASSWORD> }</pre>	<p><PASSWORD> – пароль пользователя, задается строкой до 32 символов.</p>

15	Включить пользователя.	RTT (config-l2tp-user) enable	
16	Указать список WINS-серверов, которые будут использовать удаленные пользователи (не обязательно).	RTT (config-l2tp-server) # wins-servers object-group <OBJ-GROUP-NETWORK-NAME>	<OBJ-GROUP-NETWORK-NAME> – имя профиля IP-адресов, который содержит адреса необходимых WINS-серверов, задаётся строкой до 31 символа.

7.29.2 Пример настройки

Задача:

Настроить L2TP-сервер на маршрутизаторе для подключения удаленных пользователей к ЛВС. Аутентификация пользователей проходит на RADIUS-сервере.

- адрес L2TP-сервера – 120.11.5.1;
- шлюз внутри туннеля – 10.10.10.1;
- адрес Radius-сервера – 192.168.1.4;

Для IPsec используется метод аутентификации по ключу: ключ — «password».

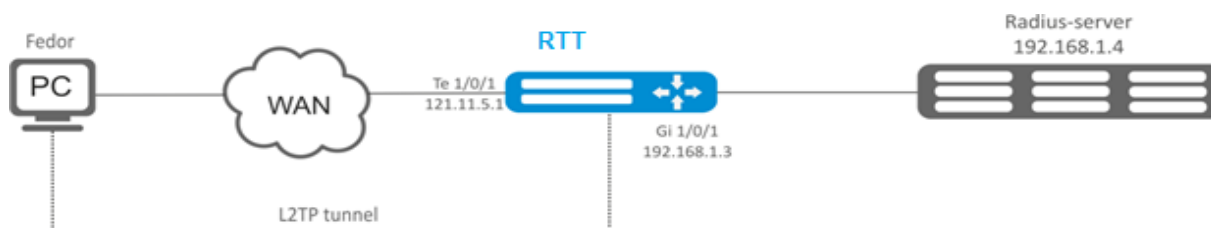


Рисунок 51 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- Настроить подключение к RADIUS-серверу;
- Настроить зоны для интерфейсов te1/0/1 и gi1/0/1;
- Указать IP-адреса для интерфейсов te1/0/1 и te1/0/1.

Создадим профиль адресов, содержащий адрес локального шлюза:

```
RTT(config)# object-group network l2tp_local
RTT(config-object-group-network)# ip address-range 10.10.10.1
RTT(config-object-group-network)# exit
```

Создадим профиль адресов, содержащий DNS-серверы:

```
RTT(config)# object-group network pptp_dns
RTT(config-object-group-network)# ip address-range 8.8.8.8
RTT(config-object-group-network)# ip address-range 8.8.4.4
RTT(config-object-group-network)# exit
```

Создадим L2TP-сервер и привяжем к нему вышеуказанные профили:


```
RTT(config)# remote-access l2tp remote-workers
RTT(config-l2tp)# local-address ip-address 10.10.10.1
RTT(config-l2tp)# remote-address address-range 10.10.10.5-10.10.10.15
RTT(config-l2tp)# outside-address ip-address 120.11.5.1
RTT(config-l2tp)# dns-server object-group l2tp_dns
```

Выберем метод аутентификации пользователей L2TP-сервера:

```
RTT(config-l2tp)# authentication mode radius
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
RTT(config-l2tp)# security-zone VPN
```

Выберем метод аутентификации первой фазы IKE и зададим ключ аутентификации:

```
RTT(config-l2tp)# ipsec authentication method psk
RTT(config-l2tp)# ipsec authentication pre-shared-key ascii-text password
```

Включим L2TP-сервер:

```
RTT(config-l2tp)# enable
```

После применения конфигурации маршрутизатор будет прослушивать IP-адрес 120.11.5.1 и порт 1701. Состояние сессий L2TP-сервера можно посмотреть командой:

```
RTT# show remote-access status l2tp server remote-workers
```

Счетчики сессий L2TP-сервера можно посмотреть командой:

```
RTT# show remote-access counters l2tp server remote-workers
```

Очистить счетчики сессий L2TP-сервера можно командой:

```
RTT# clear remote-access counters l2tp server remote-workers
```

Завершить сессию пользователя fedor L2TP-сервера можно одной из следующих команд:

```
RTT# clear remote-access session l2tp username fedor
RTT# clear remote-access session l2tp server remote-workers username fedor
```

Конфигурацию L2TP-сервера можно посмотреть командой:

```
RTT# show remote-access configuration l2tp remote-workers
```



Помимо создания L2TP-сервера необходимо в firewall открыть UDP-порты 500, 1701, 4500 для обслуживания соединений и разрешить протоколы ESP(50) и GRE(47) для туннельного трафика.

7.30 Настройка удаленного доступа к корпоративной сети по OpenVPN протоколу

OpenVPN — полнофункциональное средство для построения виртуальных частных сетей (Virtual Private Networks, VPN), организации удалённого доступа, и решения ряда других задач, связанных с безопасностью передачи данных, базирующееся на SSL.

7.30.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать профиль OpenVPN-сервера.	<code>RTT(config)# remote-access openvpn <NAME></code>	<NAME> – имя профиля OpenVPN-сервера, задаётся строкой до 31 символа.
2	Указать список IP-адресов, из которого OpenVPN сервером выдаются динамические IP-адреса удаленным пользователям в режиме L2. (только для tunnel ethernet)	<code>RTT(config-openvpn-server)# address-range <FROM-ADDR>-<TO-ADDR></code>	<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].
3	Включить клиентские соединения по OpenVPN в L2 домен (только для tunnel ethernet).	<code>RTT(config-openvpn-server)# bridge-group <BRIDGE-ID></code>	<BRIDGE-ID> – идентификационный номер моста.
4	Указать сертификаты и ключи.	<code>RTT(config-openvpn-server)# certificate <CERTIFICATE-TYPE><NAME></code>	<CERTIFICATE-TYPE> - тип сертификата или ключа, может принимать следующие значения: ca – сертификат удостоверяющего сервера; crl – список отозванных сертификатов; dh – ключ Диффи-Хеллмана; server-crt – публичный сертификат сервера; server-key – приватный ключ сервера; ta – HMAC ключ. <NAME> – имя сертификата или ключа, задаётся строкой до 31 символа.
5	Включить блокировку передачи данных между клиентами (не обязательно).	<code>RTT(config-openvpn-server)# client-isolation</code>	
6	Устанавливается максимальное количество одновременных пользовательских сессий (не обязательно).	<code>RTT(config-openvpn-server)# client-max <VALUE></code>	<VALUE> – максимальное количество пользователей, принимает значения [1..65535].
7	Включается механизм сжатия передаваемых данных между	<code>RTT(config-openvpn-server)# compression</code>	

	клиентами и сервером OpenVPN (не обязательно).		
8	Указать описание конфигурируемого сервера (не обязательно).	RTT (config-openvpn-server) # description <DESCRIPTION>	<DESCRIPTION> – описание OpenVPN-сервера, задаётся строкой до 255 символов.
9	Указать список DNS-серверов, которые будут использовать удаленные пользователи (не обязательно).	RTT (config-openvpn-server) # dns-server <ADDR>	<ADDR> – IP-адрес DNS сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];
10	Выбрать алгоритм шифрования, используемый при передаче данных.	RTT (config-openvpn-server) # encryption algorithm <ALGORITHM>	<ALGORITHM> – идентификатор протокола шифрования, принимает значения: 3des,blowfish128, aes128.
11	Определим подсеть, из которой выдаются IP-адреса пользователям. (только для tunnel ip)	RTT (config-openvpn-server) # network <ADDR/LEN>	<ADDR/LEN> – адрес 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 [1..32];
12	Указать TCP/UDP порт, который будет прослушиваться OpenVPN-сервером (не обязательно).	RTT (config-openvpn-server) # port <PORT>	<PORT> – TCP/UDP порт, принимает значения [1..65535].
13	Указать инкапсулируемый протокол.	RTT (config-openvpn-server) # protocol <PROTOCOL>	<PROTOCOL> – тип инкапсуляции, возможные значения: TCP -инкапсуляция в TCP-сегменты; UDP -инкапсуляция в UDP-дейтаграммы.
14	Включить анонсирование маршрута по умолчанию для OpenVPN соединений, что приводит к замене маршрута по умолчанию на клиентской стороне (не обязательно).	RTT (config-openvpn-server) # redirect-gateway	
16	Включить анонсирование указанных подсетей, шлюзом является IP-адрес OpenVPN-сервера (не обязательно).	RTT (config-openvpn-server) # route <ADDR/LEN>	<ADDR/LEN> – адрес 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 [1..32];
17	Включить OpenVPN-сервер в зону безопасности и настроить правила взаимодействия между зонами.	RTT (config-openvpn-server) # security-zone <NAME>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
18	Указать временной интервал, по истечению которого встречная сторона считается недоступной (не обязательно).	RTT (config-openvpn-server) # timers holdtime <TIME>	<TIME> – время в секундах, принимает значения [1..65535].
19	Указать временной интервал, по истечению которого идет проверка соединения со	RTT (config-openvpn-server) # timers keepalive <TIME>	<TIME> – время в секундах, принимает значения [1..65535].

	встречной стороной (не обязательно).		
20	Определить тип соединения с частной сетью через OpenVPN-сервер.	RTT (config-openvpn-server) # tunnel <TYPE>	<TYPE> – инкапсулирующий протокол, принимает значения: ip – соединение точка-точка; ethernet – подключение к L2 домену.
21	Определить подсеть для указанного пользователя OpenVPN-сервера (при использовании локальной базы для аутентификации пользователей).	RTT (config-openvpn-server) # username <NAME>subnet <ADDR/LEN>	<NAME> – имя пользователя, задаётся строкой до 31 символа. <ADDR/LEN> – адрес 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 [1..32].
22	Указать список WINS-серверов, которые будут использовать удаленные пользователи (не обязательно).	RTT (config-openvpn-server) # wins-server <ADDR>	<ADDR> – IP-адрес WINS сервера, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
23	Включить профиль OpenVPN-сервера.	RTT (config-openvpn-server) # enable	

7.30.2 Пример настройки

Задача:

Настроить OpenVPN-сервер в режиме L3 на маршрутизаторе для подключения удаленных пользователей к ЛВС.

- подсеть OpenVPN-сервера – 10.10.100.0/24;
- режим – L3;
- аутентификация на основе сертификатов.



Рисунок 52 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- Подготовить сертификаты и ключи:
 - Сертификат Удостоверяющего Центра (CA)
 - Ключ и сертификат для OpenVPN сервера
 - Ключ Диффи-Хелмена и HMAC для TLS

- Настроить зону для интерфейса te1/0/1
- Указать IP-адреса для интерфейса te1/0/1.

Импортируем по tftp сертификаты и ключи

```
RTT# copy tftp://192.168.16.10:/ca.crt certificate:ca/ca.crt
RTT# copy tftp://192.168.16.10:/dh.pem certificate:dh/dh.pem
RTT# copy tftp://192.168.16.10:/server.key certificate:server-key/server.key
RTT# copy tftp://192.168.16.10:/server.crt certificate:server-crt/server.crt
RTT# copy tftp://192.168.16.10:/ta.key certificate:ta/ta.key
```

Создадим OpenVPN-сервер и подсеть в которой он будет работать:

```
RTT(config)# remote-access openvpn AP
RTT(config-openvpn)# network 10.10.100.0/24
```

Укажем тип соединения L3 и протокол инкапсуляции.

```
RTT(config-openvpn)# tunnel ip
RTT(config-openvpn)# protocol tcp
```

Объявим подсети ЛВС которые будут доступны через OpenVPN соединение и укажем DNS сервер

```
RTT(config-)# route 10.10.0.0/20
RTT(config-openvpn)# dns-server 10.10.1.1
```

Укажем ранее импортированные сертификаты и ключи, которые будет использоваться OpenVPN-сервером:

```
RTT(config-openvpn)# certificate ca ca.crt
RTT(config-openvpn)# certificate dh dh.pem
RTT(config-openvpn)# certificate server-key server.key
RTT(config-openvpn)# certificate server-crt server.crt
RTT(config-openvpn)# certificate ta ta.key
```

Укажем зону безопасности, к которой будут относиться сессии пользователей:

```
RTT(config-openvpn)# security-zone VPN
```

Выберем алгоритм шифрования aes128:

```
RTT(config-openvpn)# encryption algorithm aes128
```

Включим OpenVPN-сервер:

```
RTT(config-openvpn)# enable
```

После применения конфигурации маршрутизатор будет прослушивать порт 1194 (используется по умолчанию).

Состояние сессий OpenVPN-сервера можно посмотреть командой:

```
RTT# show remote-access status openvpn server AP
```

Счетчики сессий OpenVPN-сервера можно посмотреть командой:

```
RTT# show remote-access counters openvpn server AP
```

Очистить счетчики сессий OpenVPN-сервера можно командой:

```
RTT# clear remote-access counters openvpn server AP
```

Завершить сессию пользователя fedor OpenVPN-сервера можно одной из следующих команд:

```
RTT# clear remote-access session openvpn username fedor
RTT# clear remote-access session openvpn server AP username fedor
```

Конфигурацию OpenVPN-сервера можно посмотреть командой:

```
RTT# show remote-access configuration openvpn AP
```



Помимо создания OpenVPN-сервера необходимо в firewall открыть TCP-порт 1194.

7.31 Настройка клиента удаленного доступа по протоколу PPPoE

PPPoE — это туннелирующий протокол (tunneling protocol), который позволяет инкапсулировать IP PPP через соединения Ethernet и обладает программными возможностями PPP-соединений, что позволяет использовать его для виртуальных соединений на соседнюю Ethernet-машину и устанавливать соединение точка-точка, которое используется для транспортировки IP-пакетов, а также работает с возможностями PPP. Это позволяет применять традиционное PPP-ориентированное ПО для настройки соединения, которое использует не последовательный канал, а пакетно-ориентированную сеть (например, Ethernet), чтобы организовать классическое соединение с логином и паролем для Интернет-соединений. Кроме того, IP-адрес по другую сторону соединения назначается только когда PPPoE-соединение открыто, позволяя динамическое переиспользование IP-адресов.

7.31.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать PPPoE-туннель и перейти в режим конфигурирования PPPoE-клиента.	RTT (config) # tunnel pppoe <PPPoE>	<PPPoE> – порядковый номер туннеля от 1 до 10.
2	Указать описание конфигурируемого клиента (не обязательно).	RTT (config-pppoe) # description <DESCRIPTION>	<DESCRIPTION> – описание PPPoE-сервера, задаётся строкой до 255 символов.
3	Указать метод аутентификации (не обязательно).	RTT (config-pptp) # authentication method <METHOD>	<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap Значение по умолчанию: chap
4	Включить отказ от получения маршрута по умолчанию от PPPoE-сервера (не обязательно).	RTT (config-pppoe) # ignore-default-route	

5	Указать интерфейс через который будет устанавливаться PPPoE соединение.	<code>RTT (config-pppoe) # interface <IF></code>	<IF> – интерфейс или группа интерфейсов.
6	Указать интервал времени, за который усредняется статистика о нагрузке (не обязательно).	<code>RTT (config-pppoe) # load-average <TIME></code>	<TIME> - интервал времени в секундах от 5 до 150 (по умолчанию 5 сек)
7	Указать размер MTU (MaximumTransmissionUnit) для PPPoE-туннеля. MTU более 1500 будет активно только если применена команда "system jumbo-frames" (не обязательно).	<code>RTT (config-pppoe) # mtu <MTU></code>	<MTU> – значение MTU, принимает значения в диапазоне: [1280..10000]. _x000B_ Для LT-туннелей: [552-10000]. Значение по умолчанию 1500
8	Указать имя пользователя и пароль для подключения к PPPoE-серверу	<code>RTT (config-pppoe) # username <NAME> password ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<name> – имя пользователя, задаётся строкой до 31 символа <CLEAR-TEXT> – пароль, задаётся строкой [8 .. 64] символов; <ENCRYPTED-TEXT> – зашифрованный пароль, задаётся строкой [16..128] символов.
9	Указать имя экземпляра VRF, в котором будут использоваться указанные сетевой интерфейс, мост, зона безопасности, сервер динамической авторизации (DAS) или группа правил NAT. (не обязательно)	<code>RTT (config-pppoe) # ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
10	Отключения функции Firewall на сетевом интерфейсе (не обязательно)	<code>RTT (config-pppoe) # ip firewall disable</code>	
	Настройка зоны безопасности	<code>RTT (config-pppoe) # security-zone <NAME></code>	<NAME>-имя зоны безопасности, задаётся строкой до 31 символа.
11	Активировать конфигурируемый профиль	<code>RTT (config-pppoe) # enable</code>	

7.31.2 Пример настройки PPPoE-клиента

Задача:

Настроить PPPoE-клиент на маршрутизаторе.

- Учетные записи для подключения – tester;
- Пароли учетных записей – password;
- Подключение должно осуществляться с интерфейса gigabitethernet 1/0/7.

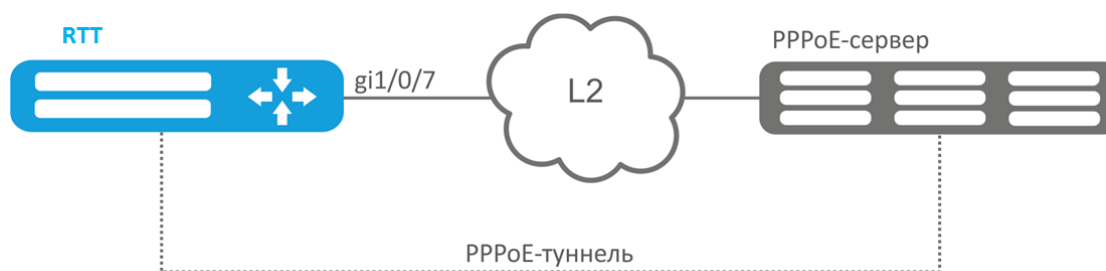


Рисунок 53 – Схема сети

Решение:

Предварительно настроить PPPoE-сервер с учетными записями.

Зайдем в режим конфигурирования PPPoE-клиента и отключим межсетевого экран:

```
RTT# configure
RTT(config)# tunnel pppoe 1
RTT(config-pppoe)# ip firewall disable
```

Укажем пользователя и пароль для подключения к PPPoE-серверу:

```
RTT(config-pppoe)# username tester password ascii-text password
```

Укажем интерфейс через который будет устанавливаться PPPoE-соединение:

```
RTT(config-pppoe)# interface gigabitethernet 1/0/7
RTT(config-pppoe)# enable
```

Состояние PPPoE-туннеля можно посмотреть командой:

```
RTT# show tunnels configuration pppoe 1
```

Счетчики сессий PPPoE-клиента можно посмотреть командой:

```
RTT# show tunnels counters pppoe 1
```

7.32 Настройка клиента удаленного доступа по протоколу PPTP

PPTP (англ. Point-to-Point Tunneling Protocol) – туннельный протокол типа точка-точка, позволяющий устанавливать защищённое соединение за счёт создания специального туннеля в обычной незащищенной сети. PPTP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. PPTP может также использоваться для организации туннеля между двумя локальными сетями. PPTP использует дополнительное TCP-соединение для обслуживания туннеля.

7.32.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать PPTP-туннель и перейти в режим его конфигурирования.	RTT(config)# tunnel pptp <INDEX>	<INDEX> – идентификатор туннеля в диапазоне: [1..10].

2	Указать метод аутентификации (не обязательно).	<code>RTT(config-pptp) # authentication method <METHOD></code>	<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap Значение по умолчанию: chap
3	Указать экземпляр VRF, в котором будет работать данный PPTP-туннель (не обязательно).	<code>RTT(config-pptp) # ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задается строкой до 31 символа.
4	Указать описание конфигурируемого туннеля (не обязательно).	<code>RTT(config-pptp) # description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задается строкой до 255 символов.
5	Установить удаленный IP-адрес для установки туннеля.	<code>RTT(config-pptp) # remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
6	Установить размер MTU (MaximumTransmissionUnit) для туннеля (не обязательно)	<code>RTT(config-pptp) # mtu <MTU></code>	<MTU> – значение MTU, принимает значение в диапазоне: [552..10000]. Значение по умолчанию: 1500
7	Игнорировать маршрут по умолчанию через данный PPTP-туннель (не обязательно)	<code>RTT(config-pptp) # ignore-default-route</code>	
8	Задать интервал времени, за который усредняется статистика о нагрузке на туннель (не обязательно).	<code>RTT(config-pptp) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150] Значение по умолчанию: 5
9	Указать пользователя и установить пароль в открытой или зашифрованной форме для аутентификации удаленной стороны.	<code>RTT(config-pptp) # username <NAME> password ascii-text { <WORD> encrypted <HEX> }</code>	<NAME> – имя пользователя, задается строкой до 31 символа. <WORD> – пароль в открытой форме, задается строкой [8..64] символов, может включать символы [0-9a-fA-F]. <HEX> – пароль в зашифрованной форме, задается строкой [16..128] символов.
10	Включить PPTP-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (не обязательно).	<code>RTT(config-pptp) # security-zone <NAME></code>	<NAME> – имя зоны безопасности, задается строкой до 31 символа.
11	Задать исключение обработки входящего трафика в Firewall (не обязательно).	<code>RTT(config-pptp) # ip firewall disable</code>	
12	Активировать туннель	<code>RTT(config-pptp) # enable</code>	

7.32.2 Пример настройки удаленного подключения по PPTP-протоколу

Задача:

Настроить PPTP-туннель на маршрутизаторе:

- адрес PPTP сервера 20.20.0.1;
- учетная запись для подключения – логин: ivan, пароль: simplepass

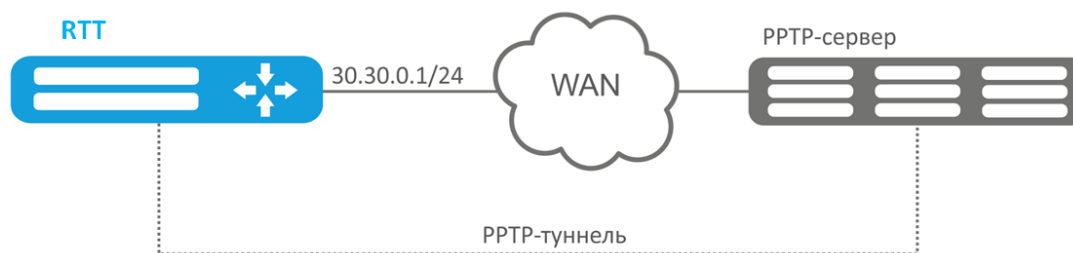


Рисунок 54 – Схема сети

Решение:

Создадим туннель PPTP:

```
RTT(config)# tunnel ptp 1
```

Укажем учетную запись (пользователя Ivan) для подключения к серверу:

```
RTT(config-ptp)# username ivan password ascii-text simplepass
```

Укажем удаленный шлюз:

```
RTT(config-ptp)# remote address 20.20.0.1
```

Укажем зону безопасности:

```
RTT(config-ptp)# security-zone VPN
```

Включим туннель PPTP:

```
RTT(config-ptp)# enable
```

Состояние туннеля можно посмотреть командой:

```
RTT# show tunnels status ptp
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
RTT# show tunnels counters ptp
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show tunnels configuration ptp
```

7.33 Настройка клиента удаленного доступа по протоколу L2TP

L2TP (англ. Layer 2 Tunneling Protocol – протокол туннелирования второго уровня) – туннельный протокол, использующийся для поддержки виртуальных частных сетей. L2TP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. L2TP может также использоваться для организации туннеля между двумя локальными сетями. L2TP использует дополнительное UDP-соединение для обслуживания туннеля. L2TP-протокол не предоставляет средств шифрования данных и поэтому он обычно используется в связке с группой протоколов IPsec, которая предоставляет безопасность на пакетном уровне.

Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать L2TP-туннель и перейти в режим его конфигурирования.	<code>RTT(config)# tunnel l2tp <INDEX></code>	<INDEX> – идентификатор туннеля в диапазоне: [1..10].
2	Указать метод аутентификации (не обязательно).	<code>RTT(config-pptp) # authentication method <METHOD></code>	<METHOD> – метод аутентификации, возможные значения: chap, mschap, mschap-v2, eap, pap Значение по умолчанию: chap
3	Указать экземпляр VRF, в котором будет работать данный L2TP-туннель (не обязательно).	<code>RTT(config-l2tp) # ip vrf forwarding <VRF></code>	<VRF> – имя VRF, задаётся строкой до 31 символа.
4	Указать описание конфигурируемого туннеля (не обязательно).	<code>RTT(config-l2tp) # description <DESCRIPTION></code>	<DESCRIPTION> – описание туннеля, задаётся строкой до 255 символов.
5	Установить удаленный IP-адрес для установки туннеля.	<code>RTT(config-l2tp) # remote address <ADDR></code>	<ADDR> – IP-адрес локального шлюза, задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
6	Установить размер MTU (MaximumTransmissionUnit) для туннеля (не обязательно)	<code>RTT(config-l2tp) # mtu <MTU></code>	<MTU> – значение MTU, принимает значение в диапазоне: [552..10000]. Значение по умолчанию: 1500
7	Игнорировать маршрут по умолчанию через данный L2TP-туннель (не обязательно)	<code>RTT(config-l2tp) # ignore-default-route</code>	
8	Задать интервал времени, за который усредняется статистика о нагрузке на туннель (не обязательно).	<code>RTT(config-l2tp) # load-average <TIME></code>	<TIME> – интервал в секундах, принимает значения [5..150] Значение по умолчанию: 5
9	Указать пользователя и установить пароль в открытой или зашифрованной форме для аутентификации удалённой стороны.	<code>RTT(config-l2tp) # username <NAME> password ascii-text { <WORD> encrypted <HEX> }</code>	<NAME> – имя пользователя, задаётся строкой до 31 символа. <WORD> – пароль в открытой форме, задаётся строкой [8..64] символов, может включать символы [0-9a-fA-F]. <HEX> – пароль в зашифрованной форме, задаётся строкой [16..128] символов.
10	Выбрать метод аутентификации по ключу для IKE-соединения.	<code>RTT(config-l2tp- server) # ipsec authentication method pre-shared- key</code>	
11	Указать общий секретный ключ для аутентификации, который должен совпадать у обеих сторон, устанавливающих туннель.	<code>RTT(config-l2tp- server) # ipsec authentication pre- shared-key { ascii- text { <TEXT> encrypted <ENCRYPTED-TEXT> } hexadecimal {<HEX> encrypted <ENCRYPTED-HEX> } }</code>	<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.

12	Включить L2TP-туннель в зону безопасности и настроить правила взаимодействия между зонами или отключить firewall (не обязательно).	<code>RTT(config-l2tp) # security-zone <NAME></code>	<NAME> – имя зоны безопасности, задаётся строкой до 31 символа.
13	Задать исключение обработки входящего трафика в Firewall (не обязательно).	<code>RTT(config-l2tp) # ip firewall disable</code>	
14	Активировать туннель	<code>RTT(config-l2tp) # enable</code>	

7.33.1 Пример настройки удаленного подключения по L2TP-протоколу

Задача:

Настроить PPTP-туннель на маршрутизаторе:

- адрес PPTP сервера 20.20.0.1;
- учетная запись для подключения – логин: ivan, пароль: simplepass



Рисунок 55 – Схема сети

Решение:

Создадим туннель L2TP:

```
RTT(config) # tunnel l2tp 1
```

Укажем учетную запись (пользователя ivan) для подключения к серверу:

```
RTT(config-l2tp) # username ivan password ascii-text simplepass
```

Укажем удаленный шлюз:

```
RTT(config-l2tp) # remote address 20.20.0.1
```

Укажем зону безопасности:

```
RTT(config-l2tp) # security-zone VPN
```

Укажем метод аутентификации ipsec:

```
RTT(config-l2tp) # ipsec authentication method pre-shared-key
```

Укажем ключ безопасности для ipsec:

```
RTT(config-l2tp) # ipsec authentication pre-shared-key ascii-text password
```

Включим туннель L2TP:

```
RTT(config-l2tp)# enable
```

Состояние туннеля можно посмотреть командой:

```
RTT# show tunnels status l2tp
```

Счетчики входящих и отправленных пакетов можно посмотреть командой:

```
RTT# show tunnels counters l2tp
```

Конфигурацию туннеля можно посмотреть командой:

```
RTT# show tunnels configuration l2tp
```

7.34 Настройка Dual-Homing ¹

Dual-Homing – технология резервирования соединений, позволяет организовать надежное соединение ключевых ресурсов сети на основе наличия резервных линков.

7.34.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Указать резервный интерфейс, на который будет происходить переключение при потере связи на основном.	<pre>RTT(config-if-gi)# backup interface<IF> vlan <VID></pre>	<IF> – интерфейс, на который будет происходить переключение <VID> – идентификационный номер VLAN, задаётся в диапазоне [2...4094]. Можно также задать диапазоном через «-» или перечислением через «,».
2	Указать количество копий пакетов с одним и тем же MAC-адресом, которые будут отправлены в активный интерфейс при переключении (не обязательно).	<pre>RTT(config)# backup- interface mac-duplicate <COUNT></pre>	<COUNT> – количество копий пакетов, принимает значение [1..4].
3	Указать количество пакетов в секунду, которое будет отправлено в активный интерфейс при переключении (не обязательно).	<pre>RTT(config)# backup- interfacemac-per- second<COUNT></pre>	<COUNT> – количество MAC-адресов в секунду, принимает значение [50..400].
4	Указать, что необходимо осуществить переключение на основной интерфейс при восстановлении связи (не обязательно).	<pre>RTT(config)# backup- interface preemption</pre>	

7.34.2 Пример настройки

Задача:

Организовать резервирование L2-соединений маршрутизатора RTT для VLAN 50-55 через устройства SW1 и SW2.

¹ В текущей версии ПО данный функционал поддерживается только на маршрутизаторе RTT-800

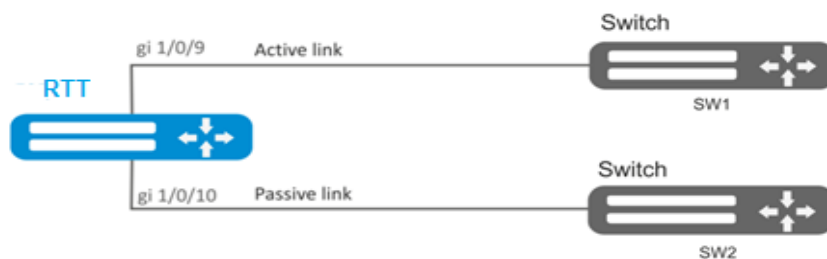


Рисунок 56 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

Создадим VLAN 50-55:

```
RTT(config) # vlan 50-55
```

Необходимо отключить STP на интерфейсах gigabitethernet 1/0/9 и gigabitethernet 1/0/10, так как совместная работа данных протоколов невозможна:

```
RTT(config) # interface gigabitethernet 1/0/9-10
```

```
RTT(config-if-gi) # spanning-tree disable
```

Интерфейсы gigabitethernet 1/0/9 и gigabitethernet 1/0/10 добавим в VLAN 50-55 в режиме general.

```
RTT(config-if-gi) # switchport general allowed vlan add 50-55
```

```
RTT(config-if-gi) # exit
```

Основной этап конфигурирования:

Сделаем интерфейс gigabitethernet 1/0/10 резервным для gigabitethernet 1/0/9:

```
RTT(config) # interface gigabitethernet 1/0/9
```

```
RTT(config-if-gi) # backup interface gigabitethernet 1/0/10 vlan 50-55
```

Просмотреть информацию о резервных интерфейсах можно командой:

```
RTT# show interfaces backup
```

7.35 Настройка QoS

QoS (Quality of Service) – технология предоставления различным классам трафика различных приоритетов в обслуживании. Использование службы QoS позволяет сетевым приложениям сосуществовать в одной сети, не уменьшая при этом пропускную способность других приложений.

7.35.1 Базовый QoS

7.35.1.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить сервис QoS на интерфейсе/туннеле/сетевом мосту.	RTT(config-if-gi) # qos enable	

	Если на интерфейсе не назначена политика QoS, то интерфейс работает в режиме BasicQoS.		
2	Установить режим доверия к значениям кодов 802.1p и DSCP во входящих пакетах. (не обязательно)	<code>RTT(config)# qos trust <MODE></code>	<p><MODE> – режим доверия к значениям кодов 802.1p и DSCP, принимает одно из следующих значений:</p> <p>dscp – режим доверия значениям кодов DSCP в IP-заголовке. Не IP-пакеты будут направлены в очередь по умолчанию.</p> <p>cos – режим доверия значениям кодов 802.1p в теге 802.1q. Нетегированные пакеты будут направлены в очередь по умолчанию.</p> <p>cos-dscp – режим доверия значениям кодов DSCP для IP-пакетов и значениям кодов 802.1p для остальных пакетов.</p>
3	Установить соответствие между значениями кодов DSCP входящих пакетов и исходящими очередями. Данное соответствие работает на входящие пакеты интерфейса/ туннеля/моста, на котором включен QOS. (не обязательно)	<code>RTT(config)# qos map dscp-queue <DSCP> to <QUEUE></code>	<p><DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63];</p> <p><QUEUE> – идентификатор очереди, принимает значения [1..8].</p> <p>Значения по умолчанию:</p> <p>DSCP: (0-7), очередь 1</p> <p>DSCP: (8-15), очередь 2</p> <p>DSCP: (16-23), очередь 3</p> <p>DSCP: (24-31), очередь 4</p> <p>DSCP: (32-39), очередь 5</p> <p>DSCP: (40-47), очередь 6</p> <p>DSCP: (48-55), очередь 7</p> <p>DSCP: (56-63), очередь 8</p>
4	Установить соответствие между значениями кодов 802.1p входящих пакетов и исходящими очередями. Данное соответствие работает на входящие пакеты интерфейса/ туннеля/моста, на котором включен QOS. (не обязательно)	<code>RTT(config)# qos map cos-queue <COS> to <QUEUE></code>	<p><COS> – классификатор обслуживания в теге 802.1q пакета, принимает значения [0..7];</p> <p><QUEUE> – идентификатор очереди, принимает значения [1..8].</p> <p>Значения по умолчанию:</p> <p>CoS: (0), очередь 1</p> <p>CoS: (1), очередь 2</p> <p>CoS: (2), очередь 3</p> <p>CoS: (3), очередь 4</p> <p>CoS: (4), очередь 5</p> <p>CoS: (5), очередь 6</p> <p>CoS: (6), очередь 7</p> <p>CoS: (7), очередь 8</p>
5	Установить соответствие между значениями кодов DSCP входящих пакетов и кодов DSCP на выходе из устройства. (в случае необходимости)	<code>RTT(config)# qos map dscp-queue <DSCP> to <DSCP></code>	<p><DSCP> – классификатор обслуживания в IP-заголовке пакета, принимает значения [0..63].</p>

	перемаркировки) Данное соответствие работает на входящие пакеты интерфейса/ туннеля/моста, на котором включен QoS.		
6	Включить изменения кодов DSCP в соответствии с таблицей DSCP-Mutation. (в случае необходимости перемаркировки)	<code>RTT(config)# qos dscp mutation</code>	
7	Установить номер очереди по умолчанию, в которую попадает весь трафик кроме IP в режиме доверия DSCP-приоритетам.	<code>RTT(config)# qos queue default <QUEUE></code>	<QUEUE> – идентификатор очереди, принимает значения [1..8].
8	Задать количество приоритетных очередей. Оставшиеся очереди являются взвешенными. (не обязательно)	<code>RTT(config)# priority-queue out num-of-queues <VALUE></code>	<VALUE> – количество очередей, принимает значение [0..8], где: 0 – все очереди участвуют в WRR (WRR – механизм обработки очередей на основе веса); 8 – все очереди обслуживаются как «strictpriority» (strictpriority – приоритетная очередь обслуживается сразу, как только появляются пакеты). Приоритетные очереди выделяются, начиная с 8-й, в сторону уменьшения номера очереди. Значение по умолчанию: 8
9	Определить веса для соответствующих взвешенных очередей.	<code>RTT(config)# qos wrr-queue <QUEUE> bandwidth <WEIGHT></code>	<QUEUE> – идентификатор очереди, принимает значение [1..8]; <WEIGHT> – значение веса, принимает значение [1..255]. Значение по умолчанию: вес 1 для всех очередей.
10	Установить ограничение скорости исходящего трафика для определенной очереди или интерфейса в целом. Команда актуальна только для BasicQoS режима интерфейса. Если трафик на входе был классифицирован при помощи расширенного QoS, ограничение не сработает. (в случае необходимости	<code>RTT(config-if-gi)# traffic-shape { <BANDWIDTH> [BURST] queue <QUEUE><BANDWIDTH> [BURST] }</code>	<QUEUE> – идентификатор очереди, принимает значение [1..8]; <BANDWIDTH> – средняя скорость трафика в Кбит/с, принимает значение [3000..10000000] для TenggabitEthernet интерфейсов и [64..1000000] для прочих интерфейсов и туннелей; <BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт. Значение по умолчанию: Отключено.

	ограничения скорости входящего потока)		
11	Установить ограничение скорости входящего трафика. (в случае необходимости ограничения скорости исходящего потока)	<pre>RTT(config-if-gi)# rate-limit <BANDWIDTH> [BURST]</pre>	<p><BANDWIDTH> – средняя скорость трафика в Кбит/с, принимает значение [3000..10000000] для TenggigabitEthernet интерфейсов и [64..1000000] для прочих интерфейсов и туннелей;</p> <p><BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт. Значение по умолчанию: Отключено.</p>

7.35.1.2 Пример настройки

Задача:

Настроить следующие ограничения на интерфейсе gigabitethernet 1/0/8: передавать трафик с DSCP 22 в первую приоритетную очередь, трафик с DSCP 14 в седьмую взвешенную очередь, установить ограничение по скорости в 60 Мбит/с для седьмой очереди.

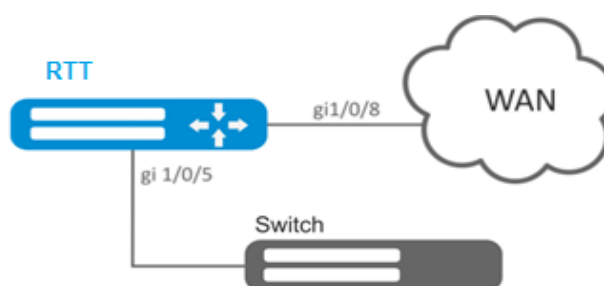


Рисунок 57 – Схема сети

Решение:

Для того чтобы первая очередь осталась приоритетной, а очереди со второй по восьмую стали взвешенными, ограничим количество приоритетных очередей до 1:

```
RTT(config)# priority-queue out num-of-queues 1
```

Перенаправим трафик с DSCP 22 в первую приоритетную очередь:

```
RTT(config)# qos map dscp-queue 22 to 1
```

Перенаправим трафик с DSCP 14 в седьмую взвешенную очередь:

```
RTT(config)# qos map dscp-queue 14 to 7
```

Включим QoS на входящем интерфейсе для корректной классификации трафика и направления в соответствующую очередь со стороны LAN:

```
RTT(config)# interface gigabitethernet 1/0/5
RTT(config-if-gi)# qos enable
RTT(config-if-gi)# exit
```

Включим QoS на интерфейсе со стороны WAN для правильной обработки очередей и ограничения полосы пропускания:

```
RTT(config)# interface gigabitethernet 1/0/8
RTT(config-if-gi)# qos enable
```

Установим ограничение по скорости в 60Мбит/с для седьмой очереди:

```
RTT(config-if)# traffic-shape queue 7 60000
RTT(config-if)# exit
```

Просмотреть статистику по QoS можно командой:

```
RTT# show qos statistics gigabitethernet 1/0/8
```

7.35.2 Расширенный QoS

7.35.2.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать списки доступа для определения трафика, к которому должен быть применен расширенный QoS.		См. Раздел Настройка списков доступа (ACL)
2	Создать класс QoS и перейти в режим настройки параметров класса.	<code>RTT(config)# class-map <NAME></code>	<NAME> – имя создаваемого класса, задается строкой до 31 символа.
3	Задать описание класса QoS. (не обязательно)	<code>RTT(config-class-map)# description <description></code>	<description> - до 255 символов.
4	Определить трафик относящийся к конфигурируемому классу по списку контроля доступа (ACL).	<code>RTT(config-class-map)# match access-group <NAME></code>	<NAME> – имя списка контроля доступа, задается строкой до 31 символа.
5	Задать значение кода DSCP, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями IP Precedence и CoS). (при необходимости перемаркировки)	<code>RTT(config-class-map)# set dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения [0..63].

6	Задать значение кода IP Precedence, которое будет установлено в IP-пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и CoS). (при необходимости перемаркировки)	RTT(config-class-map)# set ip-precedence <IPP>	<IPP> – значение кода IP Precedence, принимает значения [0..7].
7	Задать значение 802.1p приоритета, которое будет установлено в пакетах, соответствующих конфигурируемому классу (невозможно назначать одновременно с полями DSCP и IP Precedence). (при необходимости перемаркировки)	RTT(config-class-map)# set cos <COS>	<COS> – значение 802.1p приоритета, принимает значения [0..7].
8	Создать политику QoS и осуществить переход в режим настройки параметров политики.	RTT(config)# policy-map <NAME> RTT(config-policy-map)#	<NAME> – имя создаваемой политики, задается строкой до 31 символа.
9	Задать описание политики QoS. (не обязательно)	RTT(config-policy-map)# description <description>	<description> - до 255 символов.
10	Установить гарантированную полосу пропускания исходящего трафика для политики в целом.	RTT(config-policy-map)# shape average <BANDWIDTH> [BURST]	<BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [64..10000000]; <BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.
11	Включить автоматическое распределение полосы пропускания между классами, в которых нет настройки полосы пропускания, включая класс по умолчанию. (в случае необходимости)	RTT(config-policy-map)# shape auto-distribution	
12	Включить указанный QoS-класс в политику и осуществить переход в режим настройки параметров класса в рамках политики.	RTT(config-policy-map)# class <NAME> RTT(config-class-policy-map)#	<NAME> – имя привязываемого класса, задается строкой до 31 символа. При указании значения «class-default» в данный класс попадает трафик неклассифицированный на входе.
13	Включить политику QoS в класс QoS для создания иерархического QoS.	RTT(config-class-policy-map)# service-policy <NAME>	<NAME> – имя политики, задается строкой до 31 символа. Вкладываемая политика должна быть уже создана.

14	Установить гарантированную полосу пропускания исходящего трафика для класса в рамках политики. (при необходимости)	<code>RTT (config-class-policy-map) # shape average <BANDWIDTH> [BURST]</code>	<BANDWIDTH> – гарантированная полоса трафика в Кбит/с, принимает значение [64..10000000]; <BURST> – размер сдерживающего порога в КБайт, принимает значение [4..16000]. По умолчанию 128 КБайт.
15	Установить разделяемую полосу пропускания исходящего трафика для определенного класса. Данную полосу класс может занять, если менее приоритетный класс не занял свою гарантированную полосу. (при необходимости)	<code>RTT (config-class-policy-map) # shape peak <BANDWIDTH> [BURST]</code>	
16	Определить режим работы класса. (не обязательно)	<code>RTT (config-class-policy-map) # mode <MODE></code>	<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 .
17	Задать приоритет класса в WRR-процессе. (при необходимости)	<code>RTT (config-class-policy-map) # priority class <PRIORITY></code>	<PRIORITY> – приоритет класса в WRR-процессе, принимает значения [1..8]. Классы с наибольшим приоритетом обрабатываются в первую очередь.
18	Перевести класс в режим StrictPriority и задать приоритет класса. (при необходимости)	<code>RTT (config-class-policy-map) # priority level <PRIORITY></code>	<PRIORITY> – уровень приоритета в StrictPriority-процессе, принимает значения [1..8]. Классы с наибольшим приоритетом обрабатываются в первую очередь. Значение по умолчанию: класс работает в режиме WRR, приоритет не задан.
19	Определить предельное количество виртуальных очередей. (не обязательно)	<code>RTT (config-class-policy-map) # fair-queue <QUEUE-LIMIT></code>	<QUEUE-LIMIT> – предельное количество виртуальных очередей, принимает значения в диапазоне [16..4096]. Значение по умолчанию: 16.
20	Определить предельное количество пакетов для виртуальной очереди. (не обязательно)	<code>RTT (config-class-policy-map) # queue-limit <QUEUE-LIMIT></code>	<QUEUE-LIMIT> – предельное количество пакетов в виртуальной очереди, принимает значения в диапазоне [2..4096]. Значение по умолчанию: 127.

21	Определить параметры RED (Random Early Detection). (при необходимости)	<code>RTT(config-class-policy-map)# random-detect <LIMIT><MAX><MIN><PROBABILITY></code>	<p><LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].</p> <p>При указании значений должны выполняться следующие правила:</p> <p><MAX>> 2 * <MIN></p> <p><LIMIT>> 3 * <MAX></p>
22	Определить параметры GRED (Generalized Random Early Detection). (при необходимости)	<code>RTT(config-class-policy-map)# random-detect precedence <PRECEDENCE><LIMIT><MAX><MIN><PROBABILITY></code>	<p><PRECEDENCE> – значение IPPrecedence [0..7];</p> <p><LIMIT> – предельный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MAX> – максимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><MIN> – минимальный размер очереди в байтах, принимает значения в диапазоне [1..1000000];</p> <p><PROBABILITY> – вероятность отбрасывания пакетов, принимает значения [0..100].</p> <p>При указании значений должны выполняться следующие правила:</p> <p><MAX>> 2 * <MIN></p> <p><LIMIT>> 3 * <MAX></p>
23	Включить протокол компрессии tcp заголовков для трафика отдельного класса. (при необходимости)	<code>RTT(config-class-policy-map)# compression header ip tcp</code>	
24	Включить сервис QoS на интерфейсе/туннеле/сетевом мосту.	<code>RTT(config-if-gi)# qos enable</code>	
25	Назначить политику QoS на сконфигурируемом интерфейсе/туннеле/сетевом мосту для классификации входящего (input) или приоритезации исходящего (output) трафика.	<code>RTT(config-if-gi)# service-policy { input output } <NAME></code>	<p><NAME> – имя QoS-политики, задаётся строкой до 31 символа.</p>

Задача: Классифицировать входящий трафик по подсетям (10.0.11.0/24, 10.0.12.0/24), произвести маркировку по DSCP (38 и 42) и произвести разграничение по подсетям (40 Мбит/с и 60 Мбит/с), ограничить общую полосу до 250 Мбит/с, остальной трафик обрабатывать через механизм SFQ.

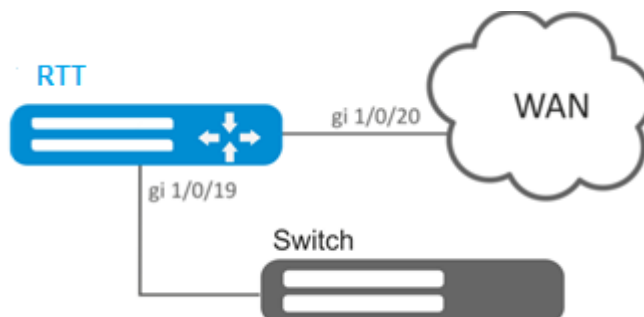


Рисунок 58 – Схема сети

Решение:

Настроим списки доступа для фильтрации по подсетям, выходим в глобальный режим конфигурации:

```
RTT(config)# ip access-list extended f11
RTT(config-acl)# rule 1
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address 10.0.11.0 255.255.255.0
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
RTT(config)# ip access-list extended f12
RTT(config-acl)# rule 1
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address 10.0.12.0 255.255.255.0
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
```

Создаем классы f11 и f12, указываем соответствующие списки доступа, настраиваем маркировку:

```
RTT(config)# class-map f11
RTT(config-class-map)# set dscp 38
RTT(config-class-map)# match access-group f11
RTT(config-class-map)# exit
RTT(config)# class-map f12
RTT(config-class-map)# set dscp 42
RTT(config-class-map)# match access-group f12
RTT(config-class-map)# exit
```

Создаём политику и определяем ограничение общей полосы пропускания:

```
RTT(config)# policy-map fl
RTT(config-policy-map)# shape average 250000
```

Осуществляем привязку класса к политике, настраиваем ограничение полосы пропускания и выходим:

```
RTT(config-policy-map)# class fl1
RTT(config-class-policy-map)# shape average 40000
RTT(config-class-policy-map)# exit
RTT(config-policy-map)# class fl2
RTT(config-class-policy-map)# shape average 60000
RTT(config-class-policy-map)# exit
```

Для другого трафика настраиваем класс с режимом SFQ:

```
RTT(config-policy-map)# class class-default
RTT(config-class-policy-map)# mode sfq
RTT(config-class-policy-map)# fair-queue 800
RTT(config-class-policy-map)# exit
RTT(config-policy-map)# exit
```

Включаем QoS на интерфейсах, политику на входе интерфейса gi 1/0/19 для классификации и на выходе gi1/0/20 для применения ограничений и режима SFQ для класса по умолчанию:

```
RTT(config)# interface gigabitethernet 1/0/19
RTT(config-if-gi)# qos enable
RTT(config-if-gi)# service-policy input fl
RTT(config-if-gi)# exit
RTT(config)# interface gigabitethernet 1/0/20
RTT(config-if-gi)# qos enable
RTT(config-if-gi)# service-policy output fl
RTT(config-if-gi)# exit
```

Для просмотра статистики используется команда:

```
RTT# do show qos policy statistics gigabitethernet 1/0/20
```

7.36 Настройка зеркалирования¹

Зеркалирование трафика — функция маршрутизатора, предназначенная для перенаправления трафика с одного порта маршрутизатора на другой порт этого же маршрутизатора (локальное зеркалирование) или на удаленное устройство (удаленное зеркалирование).

7.36.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Определить VLAN, по которому будет передаваться отзеркалированный трафик (в случае использования удаленного	RTT(config)# port monitor remote vlan <VID><DIRECTION>	<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.

¹ В текущей версии ПО данный функционал поддерживается только на маршрутизаторе RTT-RTT-R800

	зеркалирования).		
2	Включить режим удаленного зеркалирования (в случае использования удаленного зеркалирования).	<code>RTT(config)# port monitor remote</code>	
3	Определить режим порта передающего отзеркалированный трафик.	<code>RTT(config)# port monitor mode <MODE></code>	<MODE> – режим: network – совмещенный режим передачи данных и зеркалирование; monitor-only – только зеркалирование.
4	В режиме конфигурации интерфейса включить зеркалирование.	<code>RTT(config-if-gi)# port monitor interface <IF><DIRECTION></code>	<IF> – интерфейс в который будет осуществляться зеркалирование; <DIRECTION> – направление трафика: tx – зеркалирование только исходящего трафика; rx – зеркалирование только входящего трафика.

7.36.2 Пример настройки

Задача:

Организовать удаленное зеркалирование трафика по VLAN 50 с интерфейса gi1/0/11 для передачи на сервер для обработки.

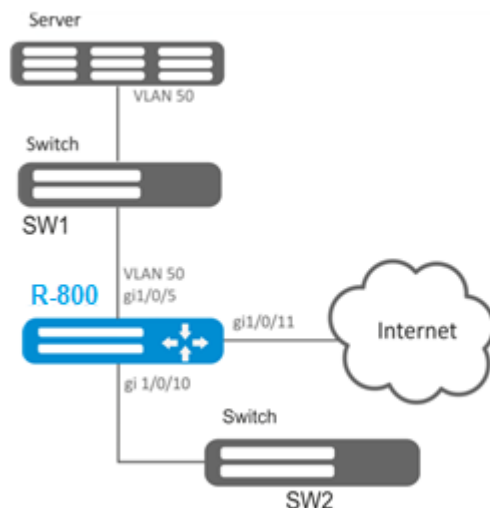


Рисунок 59 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- Создать VLAN 50;
- На интерфейсе gi 1/0/5 добавить VLAN 50 в режиме general.

Основной этап конфигурирования:

Укажем VLAN, по которой будет передаваться зеркалированный трафик:

```
RTT(config)# port monitor remote vlan 50
```

На интерфейсе gi 1/0/5 укажем порт для зеркалирования:

```
RTT(config)# interface gigabitethernet 1/0/5
RTT(config-if-gi)# port monitor interface gigabitethernet 1/0/11
```

Укажем на интерфейсе gi 1/0/5 режим удаленного зеркалирования:

```
RTT(config-if-gi)# port monitor remote
```

7.37 Настройка Netflow

Netflow — сетевой протокол, предназначенный для учета и анализа трафика. Netflow позволяет передавать данные о трафике (адрес отправителя и получателя, порт, количество информации и др.) с сетевого оборудования (сенсора) на коллектор. В качестве коллектора может использоваться обычный сервер.

7.37.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Задать версию Netflow-протокола.	<code>RTT(config)# netflow version <VERSION></code>	<VERSION> – версия Netflow-протокола: 5, 9 и 10.
2	Установить максимальное количество наблюдаемых сессий.	<code>RTT(config)# netflow max-flows <COUNT></code>	<COUNT> – количество наблюдаемых сессий, принимает значение [10000..2000000]. Значение по умолчанию: 512000.
3	Установить интервал, по истечении которого информация об устаревших сессиях экспортируются на коллектор.	<code>RTT(config)# netflow inactive-timeout <TIMEOUT></code>	<TIMEOUT> – задержка перед отправкой информации об устаревших сессиях, задается в секундах, принимает значение [0..240]. Значение по умолчанию: 15 секунд.
4	Установить частоту отправки статистики на Netflow-коллектор.	<code>RTT(config)# netflow refresh-rate <RATE></code>	<RATE> – частота отправки статистики, задается в пакетах на поток, принимает значение [1..10000]. Значение по умолчанию: 10.
5	Активировать Netflow на маршрутизаторе.	<code>RTT(config)# netflow enable</code>	
6	Создать коллектор Netflow и перейти в режим его конфигурирования.	<code>RTT(config)# netflow collector <ADDR></code>	<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
7	Установить порт Netflow-сервиса на сервере сбора статистики.	<code>RTT(config-netflow-host)# port <PORT></code>	<PORT> – номер UDP-порта, указывается в диапазоне [1..65535]. Значение по умолчанию: 2055.
8	Включить отправку статистики на Netflow-сервер в режим конфигурирования интерфейса/туннеля/сетевого моста.	<code>RTT(config-if-gi)# ip netflow export</code>	

7.37.2 Пример настройки

Задача:

Организовать учет трафика с интерфейса gi1/0/1 для передачи на сервер через интерфейс gi1/0/8 для обработки.

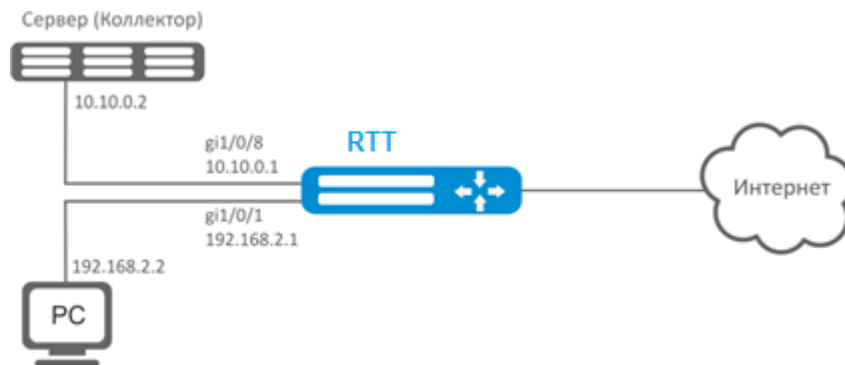


Рисунок 60 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- На интерфейсах gi1/0/1, gi1/0/8 отключить firewall командой «ip firewall disable».
- Назначить IP-адреса на портах.

Основной этап конфигурирования:

Укажем IP-адрес коллектора:

```
RTT(config) # netflow collector 10.10.0.2
```

Включим сбор экспорта статистики netflow на сетевом интерфейсе gi1/0/1:

```
RTT(config) # interface gigabitethernet 1/0/1
RTT(config-if-gi) # ip netflow export
```

Активируем netflow на маршрутизаторе:

```
RTT(config) # netflow enable
```

Для просмотра статистики Netflow используется команда:

```
RTT# show netflow statistics
```

Настройка Netflow для учета трафика между зонами аналогична настройке sFlow, описание приведено в разделе 7.38 Настройка sFlow.

7.38 Настройка sFlow

Sflow — стандарт для мониторинга компьютерных сетей, беспроводных сетей и сетевых устройств, предназначенный для учета и анализа трафика.

7.38.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Установить частоту отправки пакетов пользовательского трафика в неизменном виде на sFlow-коллектор.	<code>RTT(config)# sflow sampling-rate <RATE></code>	<RATE> – частота отправки пакетов пользовательского трафика на коллектор, принимает значение [1..10000000]. При значении частоты 10 на коллектор будет отправлен один пакет из десяти. Значение по умолчанию: 1000.
2	Установить интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса	<code>RTT(config)# sflow poll-interval <TIMEOUT></code>	<TIMEOUT> – интервал, по истечении которого происходит получение информации о счетчиках сетевого интерфейса, принимает значение [1..10000]. Значение по умолчанию: 10 секунд.
3	Активировать sFlow на маршрутизаторе.	<code>RTT(config)# sflow enable</code>	
4	Создать коллектор sFlow и перейти в режим его конфигурирования.	<code>RTT(config)# sflow collector <ADDR></code>	<ADDR> – IP-адрес коллектора, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
5	Включить отправку статистики на sFlow-сервер в режим конфигурирования интерфейса/туннеля/сетевого моста.	<code>RTT(config-if-gi)# ip sflow export</code>	

7.38.2 Пример настройки

Задача:

Организовать учет трафика между зонами trusted и untrusted.

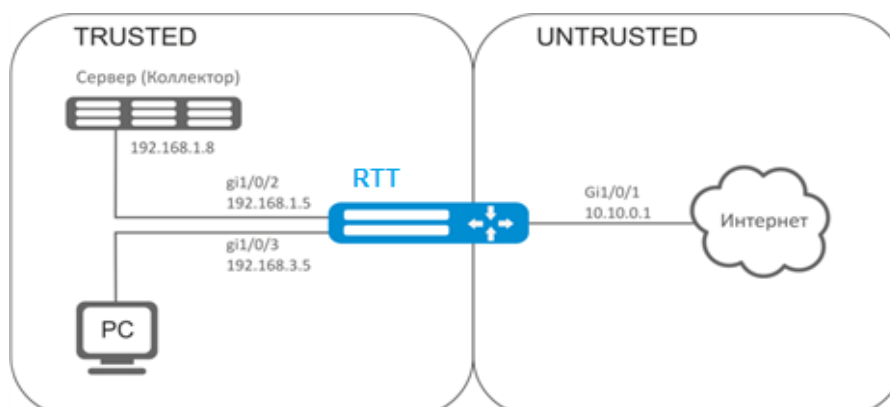


Рисунок 61 – Схема сети

Решение:

Для сетей RTT создадим две зоны безопасности:

```
RTT# configure
RTT(config)# security zone TRUSTED
RTT(config-zone)# exit
RTT(config)# security zone UNTRUSTED
RTT(config-zone)# exit
```

Настроим сетевые интерфейсы и определим их принадлежность к зонам безопасности:

```
RTT(config)# interface gil/0/1
RTT(config-if-gi)# security-zone UNTRUSTED
RTT(config-if-gi)# ip address 10.10.0.1/24
RTT(config-if-gi)# exit
RTT(config)# interface gil/0/2-3
RTT(config-if-gi)# security-zone TRUSTED
RTT(config-if-gi)# exit
RTT(config)# interface gil/0/2
RTT(config-if-gi)# ip address 192.168.1.5/24
RTT(config-if-gi)# exit
RTT(config)# interface gil/0/3
RTT(config-if-gi)# ip address 192.168.3.5/24
RTT(config-if-gi)# exit
```

Укажем IP-адрес коллектора:

```
RTT(config)# sflow collector 192.168.1.8
```

Включим экспорт статистики по протоколу sFlow для любого трафика в правиле «rule1» для направления TRUSTED-UNTRUSTED:

```
RTT(config)# security zone-pair TRUSTED UNTRUSTED
RTT(config-zone-pair)# rule 1
RTT(config-zone-pair-rule)# action sflow-sample
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
```

Активируем sFlow на маршрутизаторе:

```
RTT(config)# sflow enable
```

Настройка sFlow для учета трафика с интерфейса осуществляется аналогично 7.37 Настройка Netflow.

7.39 Настройка LACP

LACP — протокол для агрегирования каналов, позволяет объединить несколько физических каналов в один логический. Такое объединение позволяет увеличивать пропускную способность и надежность канала.

7.39.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Установить приоритет системы для протокола LACP.	RTT(config)# lacp system-priority <PRIORITY>	<PRIORITY> – приоритет, указывается в диапазоне [1..65535]. Значение по умолчанию: 1.

2	Установить механизм балансировки нагрузки для групп агрегации каналов.	<code>RTT(config)# port-channel load-balance {src-dst-mac-ip src-dst-mac src-dst-ip src-dst-mac-ip-port}</code>	<p>– src-dst-mac-ip – механизм балансировки основывается на MAC-адресе и IP-адресе отправителя и получателя;</p> <p>– src-dst-mac – механизм балансировки основывается на MAC-адресе отправителя и получателя;</p> <p>– src-dst-ip – механизм балансировки основывается на IP-адресе отправителя и получателя;</p> <p>– src-dst-mac-ip-port – механизм балансировки основывается на MAC-адресе, IP-адресе и порте отправителя и получателя.</p>
3	Установить административный таймаут протокола LACP.	<code>RTT(config)# lacp timeout { short long }</code>	<p>- long – длительное время таймаута;</p> <p>- short – короткое время таймаута.</p> <p>Значение по умолчанию: long.</p>
4	Создать и перейти в режим конфигурирования агрегированного интерфейса.	<code>RTT(config)# interface port-channel <ID></code>	<ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].
5	Настроить необходимые параметры агрегированного канала.		
6	Перейти в режим конфигурирования физического интерфейса.	<code>RTT(config)# interface <IF-TYPE><IF-NUM></code>	<p><IF-TYPE> тип интерфейса (gigabitethernet или tengigabitethernet).</p> <p><IF-NUM> - F/S/P – F-фрейм (1), S – слот (0), P – порт.</p>
7	Включить физический интерфейс в группу агрегации каналов с указанием режима формирования группы агрегации каналов.	<code>RTT(config-if-gi)# channel-group <ID> mode <MODE></code>	<p><ID> – порядковый номер группы агрегации каналов, принимает значения [1..12].</p> <p><MODE> – режим формирования группы агрегации каналов:</p> <p>– auto – добавить интерфейс в динамическую группу агрегации с поддержкой протокола LACP;</p> <p>– on – добавить интерфейс в статическую группу агрегации.</p>
8	Установить LACP-приоритет интерфейса Ethernet.	<code>RTT(config-if-gi)# lacp port-priority <PRIORITY></code>	<p><PRIORITY> – приоритет, указывается в диапазоне [1..65535].</p> <p>Значение по умолчанию: 1.</p>

7.39.2 Пример настройки

Задача:

Настроить агрегированный канал между маршрутизатором RTT и коммутатором.



Рисунок 62 – Схема сети

Решение:

Предварительно нужно выполнить следующие настройки:

На интерфейсах gi1/0/1, gi1/0/2 отключить зону безопасности командой «no security-zone».

Основной этап конфигурирования:

Создадим интерфейс port-channel 2:

```
RTT(config)# interface port-channel 2
```

Включим физические интерфейсы gi1/0/1, gi1/0/2 в созданную группу агрегации каналов:

```
RTT(config)# interface gigabitethernet 1/0/1-2
RTT(config-if-gi)# channel-group 2 mode auto
```

Дальнейшая конфигурация port-channel проводится как на обычном физическом интерфейсе.

7.40 Настройка VRRP

VRRP (Virtual Router Redundancy Protocol) — сетевой протокол, предназначенный для увеличения доступности маршрутизаторов, выполняющих роль шлюза по умолчанию. Это достигается путём объединения группы маршрутизаторов в один виртуальный маршрутизатор и назначения им общего IP-адреса, который и будет использоваться как шлюз по умолчанию для компьютеров в сети.

7.40.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Перейти в режим конфигурирования интерфейса/туннеля/ сетевого моста, для которого необходимо настроить протокол VRRP	RTT(config)# interface <IF- TYPE><IF-NUM>	<IF-TYPE> – тип интерфейса; <IF-NUM> - F/S/P – F-фрейм (1), S – слот (0), P – порт.
		RTT(config)# tunnel <TUN-TYPE><TUN-NUM>	<TUN-TYPE> – тип туннеля; <TUN-NUM> – номер туннеля.
		RTT(config)# bridge <BR-NUM>	<BR-NUM> - номер сетевого моста.
2	Настроить необходимые параметры на интерфейсе/туннеле/ сетевом мосту, включая IP-адрес		
3	Включить VRRP-процесс на	RTT(config-if-gi)# vrrp	

	IP-интерфейсе.	RTT(config-if-gi)# ipv6 vrrp	
4	Установить виртуальный IP-адрес VRRP-маршрутизатора.	RTT(config-if-gi)# vrrp ip <ADDR/LEN>	<ADDR/LEN> – виртуальный IP-адрес, 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 4 IP-адресов на интерфейс.
		RTT(config-if-gi)# ipv6 vrrp ip <IPV6-ADDR>	<IPV6-ADDR> – виртуальный IPv6-адрес, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF]. Можно указать до 8ми IPv6-адресов перечислением через запятую.
5	Установить идентификатор VRRP-маршрутизатора.	RTT(config-if-gi)# vrrp id <VRID>	<VRID> – идентификатора VRRP-маршрутизатора, принимает значения [1..255].
		RTT(config-if-gi)# ipv6 vrrp id <VRID>	
6	Установить приоритет VRRP-маршрутизатора.	RTT(config-if-gi)# vrrp priority <PR>	<PR> – приоритет VRRP-маршрутизатора, принимает значения [1..254]. Значение по умолчанию: 100.
		RTT(config-if-gi)# ipv6 vrrp priority <PR>	
7	Установить принадлежность VRRP-маршрутизатора 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.	RTT(config-if-gi)# vrrp group <GRID>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].
		RTT(config-if-gi)# ipv6 vrrp group <GRID>	
8	Установить IP-адрес, который будет использоваться в качестве IP-адреса отправителя для VRRP-сообщений.	RTT(config-if-gi)# vrrp source-ip <IP>	<IP> – IP-адрес отправителя, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		RTT(config-if-gi)# ipv6 vrrp source-ip <IPV6>	<IPV6> – IPv6-адрес отправителя, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
9	Установить интервал между отправкой VRRP-сообщений	RTT(config-if-gi)# vrrp timers advertise <TIME>	<TIME> – время в секундах, принимает значения [1..40]. Значение по умолчанию: 1 секунда.
		RTT(config-if-gi)# ipv6 vrrp timers advertise <TIME>	

10	Установить интервал, по истечении которого происходит отправка GratuitousARP сообщения(ий) при переходе маршрутизатора в состояние Master.	RTT(config-if-gi) # vrrp timers garp delay <TIME>	<TIME> – время в секундах, принимает значения [1..60]. Значение по умолчанию: 5 секунд.
11	Установить количество GratuitousARP сообщений, которые будут отправлены при переходе маршрутизатора в состояние Master.	RTT(config-if-gi) # vrrp timers garp repeat <COUNT>	<COUNT> – количество сообщений, принимает значения [1..60]. Значение по умолчанию: 5.
12	Установить интервал, по истечении которого будет происходить периодическая отправка GratuitousARP сообщения(ий) пока маршрутизатор находится в состоянии Master.	RTT(config-if-gi) # vrrp timers garp refresh <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: Периодическая отправка отключена.
13	Установить количество GratuitousARP сообщений, которые будут отправляться с периодом garprefresh пока маршрутизатор находится в состоянии Master.	RTT(config-if-gi) # vrrp timers garp refresh-repeat <COUNT>	<COUNT> – количество сообщений, принимает значения [1..60]. Значение по умолчанию: 1.
14	Определить, будет ли Backup-маршрутизатор с более высоким приоритетом пытаться перехватить на себя роль Master у текущего Master-маршрутизатора с более низким приоритетом.	RTT(config-if-gi) # vrrp preemption disable RTT(config-if-gi) # ipv6 vrrp preemption disable	
15	Установить 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.	RTT(config-if-gi) # vrrp preemption delay <TIME> RTT(config-if-gi) # ipv6 vrrp preemption delay <TIME>	<TIME> – время ожидания, определяется в секундах [1..1000]. Значение по умолчанию: 0

16	Установить пароль для аутентификации с соседом.	RTT(config-if-gi)# vrrp authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<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...).
17	Определить алгоритм аутентификации.	RTT(config-if-gi)# vrrp authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм аутентификации: - cleartext – пароль, передается открытым текстом; - md5 – пароль хешируется по алгоритму md5.
18	Задать версию VRRP-протокола.	RTT(config-if-gi)# vrrp version <VERSION>	<VERSION> – версия VRRP-протокола: 2, 3.
19	Установить режим, когда vrrp IP-адрес остается в состоянии UP вне зависимости от состояния самого интерфейса. (не обязательно)	RTT(config-if-gi)# vrrp force-up	
20	Определить задержку между установлением ipv6 vrrp состояния MASTER и началом рассылки ND сообщений.	RTT(config-if-gi)# ipv6 vrrp timers nd delay <TIME>	<TIME> – время в секундах, принимает значения [1..60]. Значение по умолчанию: 5
21	Определить период обновления информации протокола ND для ipv6 vrrp в состоянии MASTER.	RTT(config-if-gi)# ipv6 vrrp timers nd refresh <TIME>	<TIME> – время в секундах, принимает значения [1..65535]. Значение по умолчанию: 5
22	Определить количество ND сообщений отправляемых за период обновления для ipv6 vrrp в состоянии MASTER.	RTT(config-if-gi)# ipv6 vrrp timers nd refresh-repeat <NUM>	<NUM> – количество, принимает значения [1..60]. Значение по умолчанию: 0
23	Определить количество отправок ND пакетов после установки ipv6 vrrp в состоянии MASTER.	RTT(config-if-gi)# ipv6 vrrp timers nd repeat <NUM>	<NUM> – количество, принимает значения [1..60]. Значение по умолчанию: 1

7.40.2 Пример настройки 1

Задача: Организовать виртуальный шлюз для локальной сети в VLAN 50, используя протокол VRRP. В качестве локального виртуального шлюза используется IP-адрес 192.168.1.1.

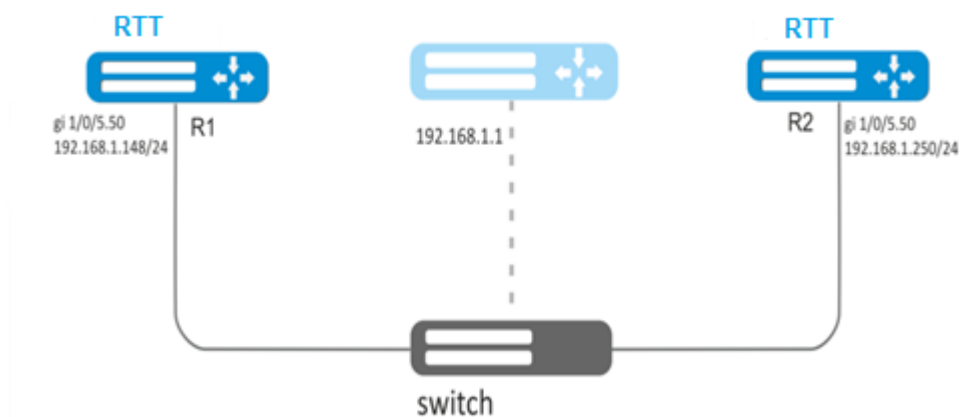


Рисунок 63 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- создать соответствующий саб-интерфейс;
- настроить зону для саб-интерфейса;
- указать IP-адрес для саб-интерфейса.

Основной этап конфигурирования:

Настроим маршрутизатор R1.

В созданном саб-интерфейсе настроим VRRP. Укажем уникальный идентификатор VRRP:

```
R1(config)#interface gi 1/0/5.50
```

```
R1(config-subif)# vrrp id 10
```

Укажем IP-адрес виртуального шлюза 192.168.1.1/24:

```
R1(config-subif)# vrrp ip 192.168.1.1
```

Включим VRRP:

```
R1(config-subif)# vrrp
```

```
R1(config-subif)# exit
```

Произвести аналогичные настройки на R2.

7.40.3 Пример настройки 2

Задача: Организовать виртуальные шлюзы для подсети 192.168.1.0/24 в VLAN 50 и подсети 192.168.20.0/24 в VLAN 60, используя протокол VRRP с функцией синхронизации Мастера. Для этого используем объединение VRRP-процессов в группу. В качестве виртуальных шлюзов используются IP-адреса 192.168.1.1 и 192.168.20.1.

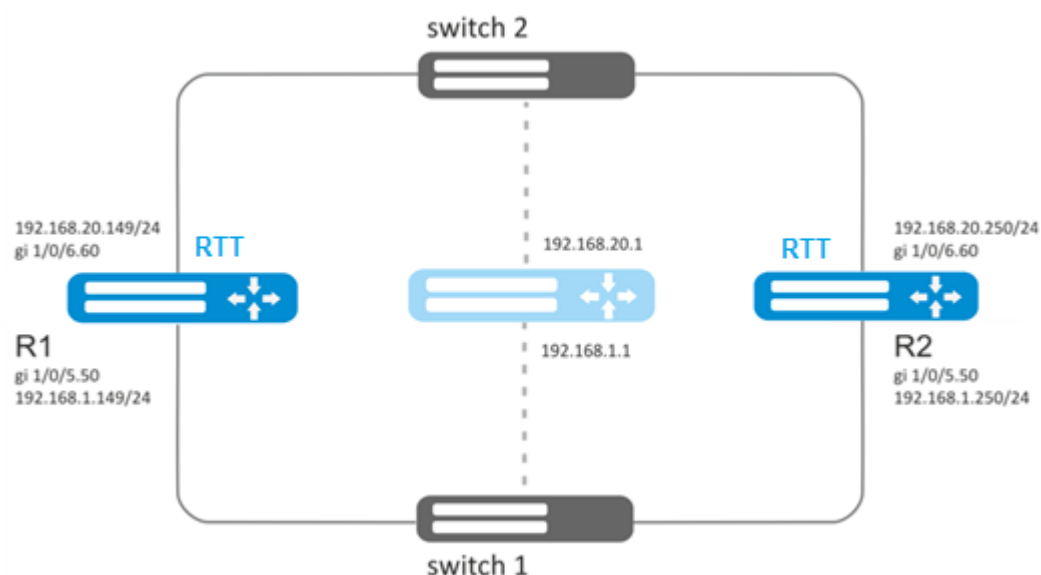


Рисунок 64 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- создать соответствующие саб-интерфейсы;
- настроить зону для саб-интерфейсов;
- указать IP-адреса для саб-интерфейсов.

Основной этап конфигурирования:

Настроим маршрутизатор R1.

Настроим VRRP для подсети 192.168.1.0/24 в созданном саб-интерфейсе.

Укажем уникальный идентификатор VRRP:

```
R1(config-sub)#interface gi 1/0/5.50
R1(config-subif)# vrrp id 10
```

Укажем IP-адрес виртуального шлюза 192.168.1.1:

```
R1(config-subif)# vrrp ip 192.168.1.1
```

Укажем идентификатор VRRP-группы:

```
R1(config-subif)# vrrp group 5
```

Включим VRRP:

```
R1(config-subif)# vrrp
R1(config-subif)# exit
```

Настроим VRRP для подсети 192.168.20.0/24 в созданном саб-интерфейсе.

Укажем уникальный идентификатор VRRP:

```
R1(config-sub)#interface gi 1/0/6.60
R1(config-subif)# vrrp id 20
```

Укажем IP-адрес виртуального шлюза 192.168.20.1:

```
R1(config-subif)# vrrp ip 192.168.20.1
```

Укажем идентификатор VRRP-группы:

```
R1(config-subif)# vrrp group 5
```

Включим VRRP:

```
R1(config-subif)# vrrp
R1(config-subif)# exit
```

Произвести аналогичные настройки на R2.



Помимо создания туннеля необходимо в firewall разрешить протокол VRRP(112).

7.41 Настройка VRRP tracking

VRRP tracking — механизм позволяющий активировать статические маршруты в зависимости от состояния VRRP.

7.41.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Настроить VRRP согласно разделу 7.6.1.		
2	Добавить в систему Tracking-объект и перейти	RTT (config) #tracking <ID>	<ID> – номер Tracking-объекта, принимает значения [1..60].

	в режим настройки параметров Tracking-объекта.		
3	Задать правило слежения за состоянием VRRP-процесса.	<code>RTT(config-tracking)# vrrp <VRID> [not] state { master backup fault }</code>	<VRID> – идентификатор отслеживаемого VRRP-маршрутизатора, принимает значения [1..255].
4	Включить Tracking-объект.	<code>RTT(config-tracking)#enable</code>	
5	Создать статический IP-маршрут к указанной подсети с указанием Tracking-объекта.	<code>RTT(config)# ip route [vrf <VRF>] <SUBNET> { <NEXTHOP> [resolve] interface <IF> tunnel <TUN> wan load-balance rule <RULE> blackhole unreachable prohibit } [<METRIC>] [track <TRACK-ID>]</code>	<p><VRF> – имя экземпляра VRF, задается строкой до 31 символа;</p> <p><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].</p> <p><NEXTHOP> – IP-адрес шлюза задается в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255];</p> <p>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 не будет установлен в систему;</p> <p><IF> – имя IP-интерфейса, задается в виде, описанном в разделе 4.2;</p> <p><TUN> – имя туннеля, задается в виде, описанном в разделе 4.3;</p> <p><RULE> – номер правила wan, задается в диапазоне [1..50];</p> <p>blackhole – при указании команды пакеты до данной подсети будут удаляться устройством без отправки уведомлений отправителю;</p> <p>unreachable – при указании команды пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Host unreachable, code 1);</p> <p>prohibit – при указании команды, пакеты до данной подсети будут удаляться устройством, отправитель получит в ответ ICMP Destination unreachable (Communication</p>

			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.
--	--	--	---

7.41.2 Пример настройки

Задача:

Для подсети 192.168.0.0/24 организован виртуальный шлюз 192.168.0.1/24 с использованием протокола VRRP на основе аппаратных маршрутизаторов R1 и R2. Так же между маршрутизаторами R1 и R2 есть линк с вырожденной подсетью 192.168.1.0/30. Подсеть 10.0.1.0/24 терминируется только на маршрутизаторе R2. ПК имеет IP-адрес 192.168.0.4/24 и шлюз по умолчанию 192.168.0.1

Когда маршрутизатор R1 находится в состоянии vrrp backup, трафик от ПК в подсеть 10.0.1.0/24 пойдет без дополнительных настроек. Когда маршрутизатор R1 находится в состоянии vrrp master, необходим дополнительный маршрут для подсети 10.0.1.0/24 через интерфейс 192.168.1.2.

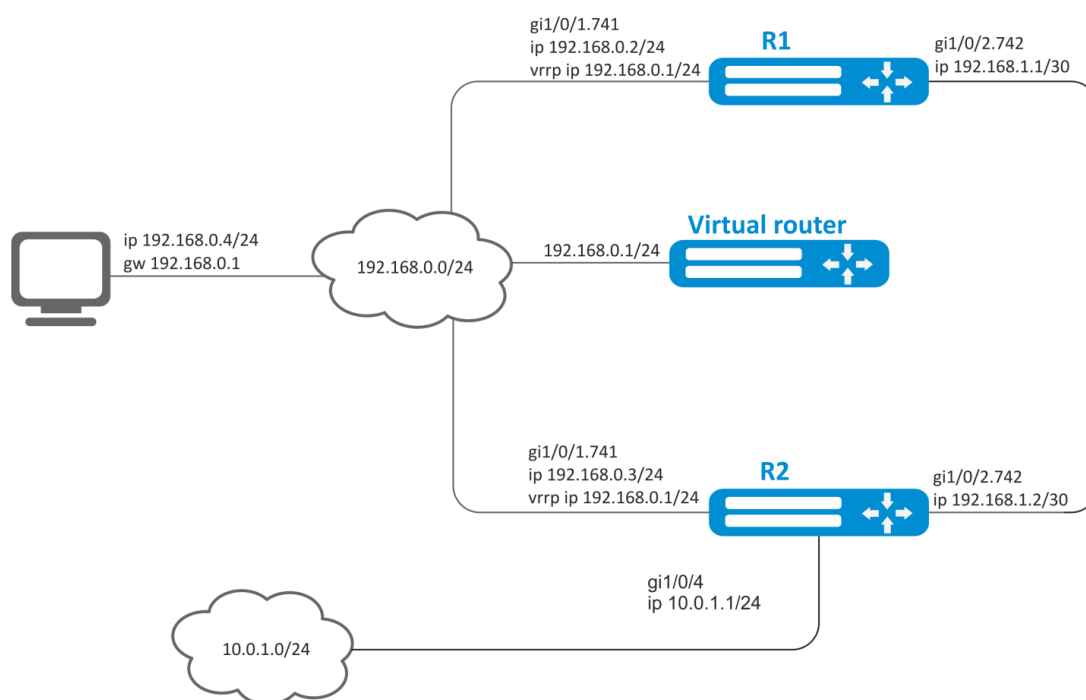


Рисунок 65 – Схема сети

Исходные конфигурации маршрутизаторов:

Маршрутизатор R1

```
hostname R1
interface gigabitethernet 1/0/1
```

```
switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
ip firewall disable
ip address 192.168.0.2/24
vrrp id 10
vrrp ip 192.168.0.1/24
vrrp
exit
interface gigabitethernet 1/0/2
switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/2.742
ip firewall disable
ip address 192.168.1.1/30
exit
```

Маршрутизатор R2

```
hostname R2
interface gigabitethernet 1/0/1
switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/1.741
ip firewall disable
ip address 192.168.0.3/24
vrrp id 10
vrrp ip 192.168.0.1/24
vrrp
exit
interface gigabitethernet 1/0/2
switchport forbidden default-vlan
exit
interface gigabitethernet 1/0/2.742
ip firewall disable
ip address 192.168.1.2/30
exit
interface gigabitethernet 1/0/4
ip firewall disable
ip address 10.0.1.1/24
exit
```

Решение:

На маршрутизаторе R2 никаких изменений не требуется так как подсеть 10.0.1.0/24 терминируется на нем, и в момент, когда R2 выступает в роли vrrp master, пакеты будут переданы в соответствующий интерфейс. На маршрутизаторе необходимо создать маршрут для пакетов с IP-адресом назначения из сети 10.0.1.0/24 в момент, когда R1 выступает в роли vrrp master.

Для этого создадим tracking-object с соответствующим условием:

```
R1(config)# tracking 1
R1(config-tracking)# vrrp 10 state master
R1(config-tracking)# enable
R1(config-tracking)# exit
```

Создадим статический маршрут в подсеть 10.0.1.0/24 через 192.168.1.2, который будет работать в случае удовлетворения условия из tracking 1:

```
R1(config)# ip route 10.0.1.0/24 192.168.1.2 track 1
```

7.42 Настройка VRF Lite

VRF (Virtual Routing and Forwarding) – технология, которая позволяет изолировать маршрутную информацию, принадлежащую различным классам (например, маршруты одного клиента).

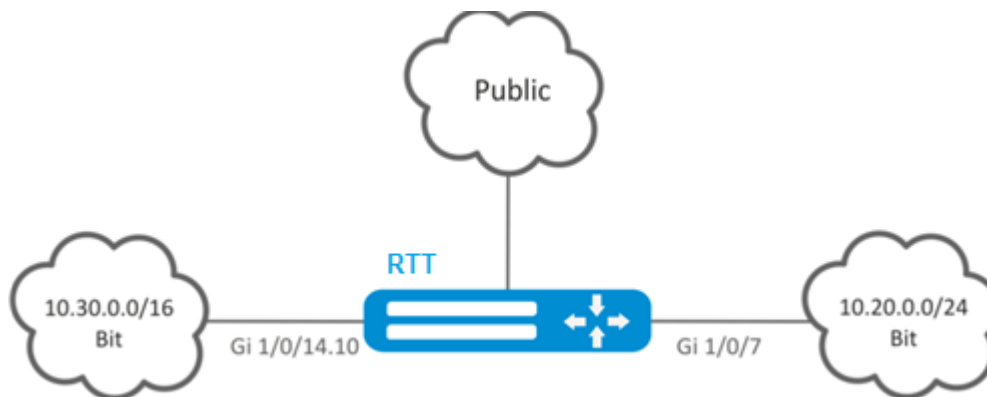


Рисунок 66 – Схема сети

7.42.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Создать экземпляр VRF и перейти в режим настройки параметров экземпляра VRF.	<code>RTT(config)# ip vrf <VRF></code>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
2	Назначить описание конфигурируемого экземпляра VRF.	<code>RTT(config-vrf)# description <DESCRIPTION></code>	<DESCRIPTION> – описание экземпляра VRF, задается строкой до 255 символов.
3	Настроить емкость таблиц маршрутизации в конфигурируемом VRF для ipv4/ipv6 протоколов маршрутизации (не обязательно).	<code>RTT(config-vrf)# ip protocols <PROTOCOL> max-routes <VALUE></code> <code>RTT(config-vrf)# ipv6 protocols <PROTOCOL> max-routes <VALUE></code>	<PROTOCOL> – вид протокола, принимает значения: ospf, bgp; <VALUE> – количество маршрутов в маршрутной таблице, принимает значения в диапазоне: OSPF RTT-RTT-R800 [1..500000], RTT-RTT-R100/RTT-R200 [1..300000]; BGP RTT-RTT-R800 [1..2600000], RTT-RTT-R100/RTT-R200 [1..1200000]. Значение по умолчанию BGP RTT-RTT-R800 (2600000), RTT-RTT-R100/RTT-R200 (1200000); OSPF RTT-RTT-R800(500000), RTT-RTT-R100/RTT-R200 (300000);
4	Включить и настроить протоколы динамической маршрутизации трафика (Static/OSPF/BGP) в экземпляре VRF (не обязательно). См. соответствующий раздел		

	7.16, 7.20 и 7.21.		
5	В режиме конфигурирования физического/логического интерфейса, туннеля, правила DNAT/SNAT, DAS-сервера или SNMPv3 пользователя указать имя экземпляра VRF для которого будет использоваться (при необходимости).	RTT(config-snat-ruleset)# ip vrf forwarding <VRF>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа.
6	Настроить LT-туннель для передачи трафика в глобальный режим или другие VRF (при необходимости)		

7.42.2 Пример настройки

Задача:

К маршрутизатору серии RTT-R подключены 2 сети, которые необходимо изолировать от остальных сетей.

Решение:

Создадим VRF:

```
RTT(config)# ip vrf bit
RTT(config-vrf)# exit
```

Создадим зону безопасности:

```
RTT(config)# security zone vrf-sec
RTT(config-zone)# ip vrf forwarding bit
RTT(config-zone)# exit
```

Создадим правило для пары зон и разрешим любой TCP/UDP-трафик:

```
RTT(config)# security zone-pair vrf-sec vrf-sec
RTT(config-zone-pair)# rule 1
RTT(config-zone-rule)# match source-address any
RTT(config-zone-rule)# match destination-address any
RTT(config-zone-rule)# match protocol udp
RTT(config-zone-rule)# match source-port any
RTT(config-zone-rule)# match destination-port any
RTT(config-zone-rule)# action permit
RTT(config-zone-rule)# enable
RTT(config-zone-rule)# exit
RTT(config-zone-pair)# rule 2
RTT(config-zone-rule)# match source-address any
RTT(config-zone-rule)# match destination-address any
RTT(config-zone-rule)# match protocol tcp
```

```
RTT(config-zone-rule) # match source-port any
RTT(config-zone-rule) # match destination-port any
RTT(config-zone-rule) # action permit
RTT(config-zone-rule) # enable
RTT(config-zone-rule) # exit
```

Создадим привязку интерфейсов, назначим IP-адреса, укажем принадлежность к зоне:

```
RTT(config) # interface gigabitethernet 1/0/7
RTT(config-if-gi) # ip vrf forwarding bit
RTT(config-if-gi) # ip address 10.20.0.1/24
RTT(config-if-gi) # security-zone vrf-sec
RTT(config-if-gi) # exit
RTT(config) # interface gigabitethernet 1/0/14.10
RTT(config-subif) # ip vrf forwarding bit
RTT(config-subif) # ip address 10.30.0.1/16
RTT(config-subif) # security-zone vrf-sec
RTT(config-subif) # exit
RTT(config) # exit
```

Информацию об интерфейсах, привязанных к VRF, можно посмотреть командой:

```
RTT# show ip vrf
```

Таблицу маршрутов VRF можно просмотреть с помощью команды:

```
RTT# show ip route vrf bit
```

7.43 Настройка MultiWAN

Технология MultiWAN позволяет организовать отказоустойчивое соединение с резервированием линков от нескольких провайдеров, а также решает проблему балансировки трафика между резервными линками.

7.43.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Сконфигурировать интерфейсы, по которым будет работать MultiWAN: установить ip-адреса и указать security-zone.		
2	Прописать статические маршруты через WAN (если необходимо).	RTT(config) # ip route <SUBNET> wan load- balance rule <ID> [<METRIC>]	<ID> – идентификатор создаваемого правила из п.2. [METRIC] – метрика маршрута, принимает значения [0..255].
3	Создать правило WAN и перейти в режим настройки параметров правила.	RTT(config) # wan load-balance rule <ID>	<ID> – идентификатор создаваемого правила, принимает значения [1..50].

4	Задать интерфейсы или туннели, которые являются шлюзами в маршруте, создаваемом службой MultiWAN.	<code>RTT (config-wan-rule) # outbound { interface <IF> tunnel <TUN> } [WEIGHT]</code>	<IF>— имя интерфейса устройства; <TUN> – имя туннеля; [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 со значением по умолчанию. В режиме резервирования активным будет маршрут с наибольшим весом. Значение по умолчанию 1.
5	Описать правила (не обязательно).	<code>RTT (config-wan-rule) # description <DESCRIPTION></code>	<DESCRIPTION> – описание правила wan, задаётся строкой до 255 символов.
6	Данной командой осуществляется переключение из режима балансировки в режим резервирования (если необходимо).	<code>RTT (config-wan-rule) # failover</code>	
7	Включить wan правило.	<code>RTT (config-wan-rule) # enable</code>	
8	Создать список IP-адресов для проверки целостности соединения и осуществить переход в режим настройки параметров списка.	<code>RTT (config) # wan load-balance target- list <NAME></code>	<NAME> – название списка, задается строкой до 31 символа.
9	Задать цель проверки и перейти в режим настройки параметров цели.	<code>RTT (config-target- list) # target <ID></code>	<ID> – идентификатор цели, задаётся в пределах [1..50]. Если при удалении используется значение параметра «all», то будут удалены все цели для конфигурируемого списка целей.
10	Описать target (не обязательно).	<code>RTT (config-wan- target) # description <DESCRIPTION></code>	<DESCRIPTION> – описание target, задаётся строкой до 255 символов.
11	Указать время ожидания ответа на запрос по протоколу ICMP (не обязательно).	<code>RTT (config-wan- target) # resp-time <TIME></code>	<TIME> – время ожидания, определяется в секундах [1..30].
12	Указать IP-адрес проверки.	<code>RTT (config-wan- target) # ip address <ADDR></code>	<ADDR> – IP-адрес назначения, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		<code>RTT (config-wan- target) # ipv6 address <IPV6-ADDR></code>	<IPV6-ADDR> – IPv6-адрес назначения, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
13	Включить проверку цели.	<code>RTT (config-wan- target) # enable</code>	
Команды для пунктов 13-17 необходимо применить на интерфейсах/туннелях в MultiWAN			

14	Включить WAN режим на интерфейсе для IPv4/IPv6 стека.	RTT(config-if-gi)# wan load-balance enable	
		RTT(config-if-gi)# ipv6 wan load-balance enable	
15	Задать количество неудачных попыток проверки соединения, после которых, при отсутствии ответа от встречной стороны, соединение будет считаться неактивным (не обязательно).	RTT(config-if-gi)# wan load-balance failure-count <VALUE>	<VALUE> – количество попыток, определяется в диапазоне [1..10]. Значение по умолчанию 1.
		RTT(config-if-gi)# ipv6 wan load-balance failure-count <VALUE>	
16	Задать количество успешных попыток проверки соединения, после которых, в случае успеха, соединение считается вновь активным (не обязательно).	RTT(config-if-gi)# wan load-balance success-count <VALUE>	<VALUE> – количество попыток, определяется в диапазоне [1..10]. Значение по умолчанию 1.
		RTT(config-if-gi)# ipv6 wan load-balance success-count <VALUE>	
17	Задать IP-адрес соседа, который будет указан в качестве одного из шлюзов в статическом маршруте, создаваемом службой MultiWAN.	RTT(config-if-gi)# wan load-balance nexthop { <IP> dhcp enable tunnel enable }	<IP> – 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 - p-t-p адрес назначения. Применимо для подключаемых интерфейсов работающих через ppp.
		RTT(config-if-gi)# ipv6 wan load-balance nexthop { <IPV6> }	
18	Данной командой будут проверяться IP-адреса из списка проверки целостности. В случае недоступности одного из проверяемых узлов, шлюз будет считаться недоступным.	RTT(config-if-gi)# wan load-balance target-list { check- all <NAME> }	<NAME> – проверку производить на основании конкретного target листа (заданного в п.7). check-all – проверку производить на основании всех target листа.
		RTT(config-if-gi)# ipv6 wan load-balance target-list { check- all <NAME> }	
19	Прописать статические маршруты через WAN (если необходимо).	RTT(config)# ip route <SUBNET> wan load- balance rule <ID> [<METRIC>]	<ID> – идентификатор создаваемого правила из п.2. [METRIC] – метрика маршрута, принимает значения [0..255].
		RTT(config)# ipv6 route <SUBNET> wan load-balance rule <ID> [<METRIC>]	

7.43.2 Пример настройки

Задача:

Настроить маршрут к серверу (108.16.0.1/28) с возможностью балансировки нагрузки.



Рисунок 67 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- настроить зоны для интерфейсов te1/0/1 и te1/0/2;
- указать IP-адреса для интерфейсов te1/0/1 и te1/0/2.

Основной этап конфигурирования:

Настроим маршрутизацию:

```
RTT(config) # ip route 108.16.0.0/28 wan load-balance rule 1
```

Создадим правило WAN:

```
RTT(config) # wan load-balance rule 1
```

Укажем участвующие интерфейсы:

```
RTT(config-wan-rule) # outbound interface tengigabitethernet 1/0/2
```

```
RTT(config-wan-rule) # outbound interface tengigabitethernet 1/0/1
```

Включим созданное правило балансировки и выйдем из режима конфигурирования правила:

```
RTT(config-wan-rule) # enable
```

```
RTT(config-wan-rule) # exit
```

Создадим список для проверки целостности соединения:

```
RTT(config) # wan load-balance target-list google
```

Создадим цель проверки целостности:

```
RTT(config-target-list) # target 1
```

Зададим адрес для проверки, включим проверку указанного адреса и выйдем:

```
RTT(config-wan-target)# ip address 8.8.8.8
RTT(config-wan-target)# enable
RTT(config-wan-target)# exit
```

Настроим интерфейсы. В режиме конфигурирования интерфейса te1/0/1 указываем nexthop:

```
RTT(config)# interface tengigabitethernet 1/0/1
RTT(config-if)# wan load-balance nexthop 203.0.0.1
```

В режиме конфигурирования интерфейса te1/0/1 указываем список целей для проверки соединения:

```
RTT(config-if)# wan load-balance target-list google
```

В режиме конфигурирования интерфейса te1/0/1 включаем WAN-режим и выходим:

```
RTT(config-if)# wan load-balance enable
RTT(config-if)# exit
```

В режиме конфигурирования интерфейса te1/0/2 указываем nexthop:

```
RTT(config)# interface tengigabitethernet 1/0/2
RTT(config-if)# wan load-balance nexthop 65.6.0.1
```

В режиме конфигурирования интерфейса te1/0/2 указываем список целей для проверки соединения:

```
RTT(config-if)# wan load-balance target-list google
```

В режиме конфигурирования интерфейса te1/0/2 включаем WAN-режим и выходим:

```
RTT(config-if)# wan load-balance enable
RTT(config-if)# exit
```

Для переключения в режим резервирования настроим следующее:

Заходим в режим настройки правила WAN:

```
RTT(config)# wan load-balance rule 1
```

Функция MultiWAN также может работать в режиме резервирования, в котором трафик будет направляться в активный интерфейс с наибольшим весом. Включить данный режим можно следующей командой:

```
RTT(config-wan-rule)# failover
```

7.44 Настройка SNMP

SNMP (англ. Simple Network Management Protocol — простой протокол сетевого управления) — протокол, предназначенный для управления устройствами в IP-сетях на основе архитектур TCP/UDP. SNMP предоставляет данные для управления в виде переменных, описывающих конфигурацию управляемой системы.

7.44.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Включить SNMP-сервер	<code>RTT(config)# snmp-server</code>	
2	Определить community для доступа по протоколу SNMPv2c.	<code>RTT(config)# snmp-server community <COMMUNITY> [<TYPE>] [{ <IP-ADDR> <IPv6-ADDR> } [view <VIEW-NAME>] [vrf <VRF>]</code>	<p><COMMUNITY> – сообщество для доступа по протоколу SNMP;</p> <p><TYPE> – уровень доступа:</p> <p>ro – доступ только для чтения;</p> <p>rw – доступ для чтения и записи.</p> <p><IP-ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].</p> <p><IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF];</p> <p><VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа;</p> <p><VRF> – имя экземпляра VRF, из которого будет разрешен доступ, задается строкой до 31 символа.</p>
3	Устанавливает значение переменной SNMP, содержащей контактную информацию	<code>RTT(config)# snmp-server contact <CONTACT></code>	<CONTACT> – контактная информация, задается строкой до 255 символов.
4	Установить значение кода DSCP для использования в IP-заголовке исходящих пакетов SNMP-сервера (не обязательно).	<code>RTT(config)# snmp-server dscp <DSCP></code>	<DSCP> – значение кода DSCP, принимает значения в диапазоне [0..63]. Значение по умолчанию: 63.
5	Разрешить перезагрузку маршрутизатора при помощи snmp-сообщений (не обязательно)	<code>RTT(config)# snmp-server system-shutdown</code>	
6	Создать SNMPv3-пользователь.	<code>RTT(config)# snmp-server user <NAME></code>	<NAME> – имя пользователя, задаётся строкой до 31 символа.
7	Устанавливает значение переменной SNMP, содержащей информацию о расположении оборудования	<code>RTT(config)# snmp-server location <LOCATION></code>	<LOCATION> – информация о расположении оборудования, задается строкой до 255 символов.
8	Определить уровень доступа пользователя по протоколу SNMPv3.	<code>RTT(config-snmp-user)# access <TYPE></code>	<p><TYPE> – уровень доступа:</p> <p>ro – доступ только для чтения;</p> <p>rw – доступ для чтения и записи.</p>
9	Определить режим безопасности	<code>RTT(config-snmp-user)# authentication access</code>	<TYPE> – режим безопасности: auth – используется только

	пользователя по протоколу SNMPv3.	<TYPE>	аутентификация; priv – используется аутентификация и шифрование данных.
10	Определить алгоритм аутентификации SNMPv3-запросов.	RTT(config-snmp-user) # authentication algorithm <ALGORITHM>	<ALGORITHM> – алгоритм шифрования: md5 – пароль шифруется по алгоритму md5; sha1 – пароль шифруется по алгоритму sha1.
11	Установить пароль для аутентификации SNMPv3-запросов.	RTT(config-snmp-user) # authentication key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<CLEAR-TEXT> – пароль, задаётся строкой от 8 до 16 символов; encrypted – при указании команды задается зашифрованный пароль: <ENCRYPTED-TEXT> – зашифрованный пароль размером от 8 байт до 16 байт (от 16 до 32 символов) в шестнадцатеричном формате (0xYYYY...) или (YYYY...).
12	Активировать фильтрацию и установить профиль IP-адресов, с которых могут приниматься SNMPv3 пакеты с данным именем SNMPv3 пользователя.	RTT(config-snmp-user) # client-list <NAME>	<NAME> - имя ранее созданной object-group, задается строкой до 31 символа.
13	Активировать фильтрацию и установить IPv4/IPv6-адрес, которому предоставлен доступ к маршрутизатору под данным SNMPv3-пользователем.	RTT(config-snmp-user) # ip address <ADDR>	<ADDR> – IP-адрес клиента, которому предоставлен доступ, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
		RTT(config-snmp-user) # ipv6 address <ADDR>	<IPv6-ADDR> – IPv6-адрес клиента, задаётся в виде X:X:X:X::X, где каждая часть принимает значения в шестнадцатеричном формате [0..FFFF].
14	Активировать SNMPv3-пользователя.	RTT(config-snmp-user) # enable	Значение по умолчанию: процесс выключен.
15	Определить алгоритм шифрования передаваемых данных.	RTT(config-snmp-user) # privacy algorithm <ALGORITHM>	<ALGORITHM> – алгоритм шифрования: aes128 – использовать алгоритм шифрования AES-128; des – использовать алгоритм шифрования DES.
16	Установить пароль для шифрования передаваемых данных.	RTT(config-snmp-user) # privacy key ascii-text { <CLEAR-TEXT> encrypted <ENCRYPTED-TEXT> }	<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...).
	Установить профиль snmp view, позволяющий разрешать или	RTT(config-snmp-user) # view <VIEW-NAME>	<VIEW-NAME> – имя SNMP view профиля, на основании которого обеспечивается доступ к OID, задается строкой до 31 символа.

	запрещать доступ к тем или иным OID для user.		
17	Включить передачу SNMP уведомлений на указанный IP-адрес и перейти в режим настройки SNMP уведомлений.	RTT(config)# snmp-server host { <IP-ADDR> <IPv6-ADDR> } [vrf <VRF>]	<IP-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.
18	Определить порт коллектора SNMP уведомлений на удаленном сервере (не обязательно).	RTT(config-snmp-host)# port <PORT>	<PORT> – номер UDP-порта, указывается в диапазоне [1..65535]. Значение по умолчанию: 162.
19	Установить фильтрацию на отправляемые SNMP-уведомления.	RTT(config)# snmp-server filter <TYPE>	<TYPE> - тип фильтруемых сообщений. Может принимать значения: bras, config, environment, files-operations, interfaces, links. Дополнительные параметры зависят от типа фильтра. См. Справочник команд CLI.
20	Создать профиль snmp view, позволяющий разрешать или запрещать доступ к тем или иным OID для community (SNMPv2) и user (SNMPv3).	RTT(config)# snmp-server view <VIEW-NAME>	<VIEW-NAME> – имя профиля SNMP view, задаётся строкой до 31 символа.

7.44.2 Пример настройки

Задача:

Настроить SNMPv3 сервер с аутентификацией и шифрованием данных для пользователя admin. IP-адрес маршрутизатора RTT - 192.168.52.41, IP-адрес сервера – 192.168.52.8.



Рисунок 68 – Схема сети

Решение:

Предварительно нужно выполнить следующие действия:

- указать зону для интерфейса gi1/0/1;
- настроить IP-адрес для интерфейсов gi1/0/1.

Основной этап конфигурирования:

Включаем SNMP-сервер:

```
RTT(config) # snmp-server
```

Создаем пользователя SNMPv3:

```
RTT(config) # snmp-server user admin
```

Определим режим безопасности:

```
RTT(snmp-user) # authentication access priv
```

Определим алгоритм аутентификации для SNMPv3-запросов:

```
RTT(snmp-user) # authentication algorithm md5
```

Установим пароль для аутентификации SNMPv3-запросов:

```
RTT(snmp-user) # authentication key ascii-text 123456789
```

Определим алгоритм шифрования передаваемых данных:

```
RTT(snmp-user) # privacy algorithm aes128
```

Установим пароль для шифрования передаваемых данных:

```
RTT(snmp-user) # privacy key ascii-text 123456789
```

Активируем SNMPv3-пользователя:

```
RTT(snmp-user) # enable
```

Определяем сервер-приемник Trap-PDU сообщений:

```
RTT(config) # snmp-server host 192.168.52.41
```

7.45 Настройка BRAS (Broadband Remote Access Server)

7.45.1 Алгоритм настройки

Шаг	Описание	Команда	Ключи
1	Добавить RADIUS-сервер в список используемых	RTT(config) # radius - server host { <IP- ADDR> <IPv6-ADDR> }	<IP-ADDR> – IP-адрес RADIUS-сервера, задаётся в виде AAA.BBB.CCC.DDD, где

	серверов и перейти в режим его конфигурирования.	<code>[vrf <VRF>]RTT(config-radius-server)#</code>	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] <VRF> – имя экземпляра VRF, задается строкой до 31 символа.
2	Задать пароль для аутентификации на удаленном RADIUS-сервере.	<code>RTT(config-radius-server)# key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT> – зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
3	Создать профиль AAA.	<code>RTT(config)# aaa radius-profile <NAME></code>	<NAME> – имя профиля сервера, задается строкой до 31 символа.
4	В профиле AAA указать RADIUS-сервер.	<code>RTT(config-aaa-radius-profile)# radius-server host { <IP-ADDR> <IPV6-ADDR> }</code>	<IP-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].
5	Создать DAS-сервер.	<code>RTT(config)# das-server <NAME></code>	<NAME> – имя DAS-сервера, задается строкой до 31 символа.
6	Задать пароль для аутентификации на удаленном DAS-сервере.	<code>RTT(config-das-server)# key ascii-text {<TEXT> encrypted <ENCRYPTED-TEXT> }</code>	<TEXT> – строка [8..16] ASCII-символов; <ENCRYPTED-TEXT>—зашифрованный пароль, размером [8..16] байт, задаётся строкой [16..32] символов.
7	Создать AAA DAS-профиль.	<code>RTT(config)# aaa das-profile <NAME></code>	<NAME> – имя DAS-профиля, задается строкой до 31 символа.
8	Указать DAS-сервер в DAS-профиле.	<code>RTT(config-aaa-das-profile)# das-server <NAME></code>	<NAME> – имя DAS-сервера, задается строкой до 31 символа.
9	Сконфигурировать BRAS.	<code>RTT(config)# subscriber-control [vrf <VRF>]</code>	<VRF> – имя экземпляра VRF, задается строкой до 31 символа, в рамках которого будет работать контроль пользователей.
10	Выбрать профиль серверов динамической авторизации (DAS), на которые будут приходить CoA-запросы от PCRF	<code>RTT(config-subscriber-control)# aaa das-profile <NAME></code>	<NAME> – имя профиля серверов динамической авторизации (DAS), задается строкой до 31 символа.
11	Выбрать профиль RADIUS-серверов для получения параметров сервисов пользователя	<code>RTT(config-subscriber-control)# aaa services-radius-profile <NAME></code>	<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.
12	Выбрать профиль RADIUS-серверов для получения параметров сессии пользователя.	<code>RTT(config-subscriber-control)# aaa sessions-radius-profile <NAME></code>	<NAME> – имя профиля RADIUS-серверов, задается строкой до 31 символа.

13	Определить IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых RADIUS пакетах.	<code>RTT(config-subscriber-control)# nas-ip-address <ADDR></code>	<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
14	Включить аутентификацию сессий по MAC-адресу (не обязательно).	<code>RTT(config-subscriber-control)# session mac-authentication</code>	
15	Организовать прозрачное пропускание служебного трафика (DHCP, DNS и т.д.) на основе фильтров.	<code>RTT(config-subscriber-control)# bypass-traffic-acl <NAME></code>	<NAME> – имя привязываемого ACL, задаётся строкой до 31 символа.
16	Перейти в режим конфигурирования сервиса по умолчанию.	<code>RTT(config-subscriber-control)# default-service</code>	
17	Привязать указанный QoS-класс к сервису по умолчанию.	<code>RTT(config-subscriber-default-service)# class-map <NAME></code>	<NAME> – имя привязываемого класса, задаётся строкой до 31 символа.
18	Указать имя списка URL, который будет использоваться для фильтрации HTTP/HTTPS трафика не аутентифицированных пользователей.	<code>RTT(config-subscriber-default-service)# filter-name { local<LOCAL-NAME> remote<REMOTE-NAME> }</code>	<LOCAL-NAME> – имя профиля URL, задаётся строкой до 31 символа; <REMOTE-NAME> – имя списка URL на удаленном сервере, задаётся строкой до 31 символа.
19	Указать действия, которые должны быть применены для HTTP/HTTPS-пакетов, URL которых входит в список URL, назначенных командой «filter-name».	<code>RTT(config-subscriber-default-service)# filter-action<ACT></code>	<ACT> – назначаемое действие: permit – прохождение трафика разрешается; deny – прохождение трафика запрещается; redirect <URL> – будет выполнен редирект на указанный URL, задаётся строкой до 255 символов.
20	Указать действия, которые должны быть применены для HTTP/HTTPS-пакетов, URL которых не входит в список URL, назначенных командой «filter-name».	<code>RTT(config-subscriber-default-service)# default-action<ACT></code>	<ACT> – назначаемое действие: permit – прохождение трафика разрешается; deny – прохождение трафика запрещается; redirect <URL> – будет выполнен редирект на указанный URL, задаётся строкой до 255 символов.
21	Активировать профиль контроля пользователей.	<code>RTT(config-subscriber-control)# enable</code>	
22	Изменить идентификатор сетевого интерфейса	<code>RTT(config-if)# location <ID></code>	<ID> – идентификатор сетевого интерфейса, задаётся строкой до 220 символов.

	(физического, саб-интерфейса или сетевого моста) (не обязательно).		
23	Включить контроль пользователей на интерфейсе.	<code>RTT(config-if-gi)# service-subscriber-control {any object-group <NAME>}</code>	<NAME> – имя профиля IP-адресов, задаётся строкой до 31 символа.
24	Включить перезапрос значения квоты при ее истечении для сервисов пользователя с настроенным ограничением по объему трафика или времени (не обязательно).	<code>RTT(config-subscriber-control)# quota-expired-reauth</code>	
25	Включить аутентификацию сессий по IP-адресу. (не обязательно)	<code>RTT(config-subscriber-control)# session ip-authentication</code>	
26	Включить прозрачное пропускание трафика в состоянии backup для BRAS (не обязательно).	<code>RTT(config-subscriber-control)# backup traffic-processing transparent</code>	
27	Задать интервал, по истечении которого с устройства будут удалены неиспользуемые в текущий момент списки URL (не обязательно).	<code>RTT(config)# subscriber-control unused-filters-remove-delay <DELAY></code>	<DELAY> – временной интервал в секундах, принимает значения [10800..86400].
28	Задать интервал, по истечении которого, если не было пакетов от пользователя, сессия считается устаревшей и удаляется с устройства (не обязательно).	<code>RTT(config-subscriber-default-service)#session-timeout <SEC></code>	<SEC> – период времени в секундах, принимает значения [120..3600].
29	Определить VRRP-группу, на основе которой определяется состояние сервиса контроля абонентов (основной/резервный) (не обязательно).	<code>RTT(config-subscriber-control)# vrrp-group <GRID></code>	<GRID> – идентификатор группы VRRP-маршрутизатора, принимает значения [1..32].
30	Определить с каких TCP-портов назначения трафик будет перенаправлен на HTTP	<code>RTT(config-subscriber-control)# ip proxy http listen-ports <NAME></code>	<NAME> – имя профиля TCP/UDP-портов, задаётся строкой до 31 символа.

	Прoxy-сервер маршрутизатора (не обязательно).		
31	Определить порт HTTP Прoxy-сервера на маршрутизаторе (не обязательно).	<code>RTT(config-subscriber-control)# ip proxy http redirect-port <PORT></code>	<PORT> – номер порта, указывается в диапазоне [1..65535].
32	Определить с каких TCP-портов назначения трафик будет перенаправлен на HTTPS Прoxy-сервер маршрутизатора (не обязательно).	<code>RTT(config-subscriber-control)# ip proxy https listen-ports <NAME></code>	<NAME> – имя профиля TCP/UDP-портов, задаётся строкой до 31 символа.
33	Определить порт HTTPS Прoxy-сервера на маршрутизаторе (не обязательно).	<code>RTT(config-subscriber-control)# ip proxy https redirect-port <PORT></code>	<PORT> – номер порта, указывается в диапазоне [1..65535].
34	Определить IP-адрес маршрутизатора, который будет использоваться в качестве IP-адреса источника в отправляемых Прoxy-сервером HTTP/HTTPS пакетах (не обязательно).	<code>RTT(config-subscriber-control)# ip proxy source-address <ADDR></code>	<ADDR> – IP-адрес источника, задаётся в виде AAA.BBB.CCC.DDD, где каждая часть принимает значения [0..255].
35	Задать URL-адрес сервера, предоставляющего списки приложений для фильтрации трафика (не обязательно)	<code>RTT(config)# subscriber-control apps-server-url <URL></code>	<URL> – адрес ссылки, задаётся строкой от 8 до 255 символов.
36	Включить контроль приложений на интерфейсе (не обязательно)	<code>RTT(config-if-gi)# subscriber-control application-filter <NAME></code>	<NAME> – имя профиля приложений, задаётся строкой до 31 символа.
37	Установить/ сбросить верхнюю границу количества сессий BRAS (не обязательно)	<code>RTT(config-subscriber-control)# thresholds sessions-number high <Threshold></code>	<Threshold> – количество сессий BRAS, [0-10000] – для RTT-R800 [0-1000] – для RTT-R100/200
38	Установить/ сбросить нижнюю границу количества сессий BRAS (не обязательно)	<code>RTT(config-subscriber-control)# thresholds sessions-number low <Threshold></code>	<Threshold> – количество сессий BRAS, [0-10000] – для RTT-R800 [0-1000] – для RTT-R100/200

7.45.2 Пример настройки BRAS с использованием ПО SoftWLC

Задача:

Предоставлять доступ до ресурсов сети Интернет, только для авторизованных пользователей.

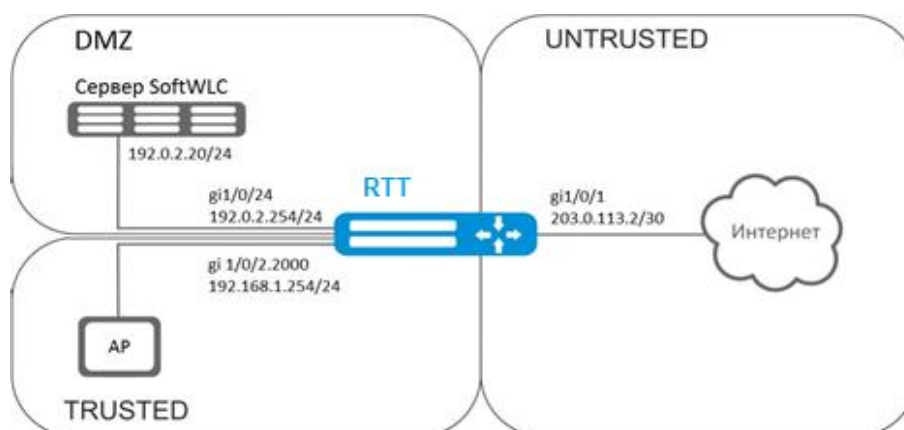


Рисунок 69 – Схема сети

Решение:

За хранение учетных данных пользователей и параметров тарифных планов отвечает сервер SoftWLC.

Для маршрутизатора необходимо наличие лицензии BRAS, после ее активации можно переходить к конфигурированию устройства.

Создадим три зоны безопасности на устройстве, согласно схеме сети:

```
RTT# configure
RTT(config)# security zone trusted
RTT(config-zone)# exit
RTT(config)# security zone untrusted
RTT(config-zone)# exit
RTT(config)# security zone dmz
RTT(config-zone)# exit
```

Сконфигурируем параметры публичного порта и сразу пропишем шлюз по умолчанию:

```
RTT(config)# interface gigabitethernet 1/0/1
RTT(config-if-gi)# security-zone untrusted
RTT(config-if-gi)# ip address 203.0.113.2/30
RTT(config-if-gi)# service-policy dynamic upstream
RTT(config-if-gi)# exit
RTT(config)# ip route 0.0.0.0/0 203.0.113.1
```

Сконфигурируем порт в сторону сервера SoftWLC:

```
RTT (config)# interface gigabitethernet 1/0/24
RTT (config-if-gi)# security-zone dmz
RTT (config-if-gi)# ip address 192.0.2.1/24
RTT (config-if-gi)# exit
```

Сконфигурируем порт для подключения Wi-Fi точки доступа.

```
RTT(config)# bridge 2
RTT(config-bridge)# security-zone trusted
RTT(config-bridge)# ip address 192.168.0.254/24
```

```
RTT(config-bridge)# ip helper-address 192.0.2.20
RTT(config-bridge)# service-subscriber-control object-group users
RTT(config-bridge)# location ssid1
RTT(config-bridge)# enable
RTT(config-bridge)# exit
RTT(config)# interface gigabitethernet 1/0/2.2000
RTT(config-subif)# bridge-group 1
RTT(config-subif)# exit
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# service-policy dynamic downstream
RTT (config-if-gi)# exit
```



Подключать клиентов необходимо через сабинтерфейсы в бриджи, причем от параметра location (смотри конфигурацию bridge 2) зависит выбор тарифного плана.

Модуль, отвечающий за AAA-операции, основан на rustel-radius и доступен по IP-адресу сервера SoftWLC. Номера портов для аутентификации и аккаунтинга в нашем примере – это значения по умолчанию для SoftWLC.

Зададим параметры для взаимодействия с этим модулем:

```
RTT(config)# radius-server host 192.0.2.20
RTT(config-radius-server)# key ascii-text password
RTT(config-radius-server)# auth-port 31812
RTT (config-radius-server)# acct-port 31813
RTT (config-radius-server)# exit
```

Создадим профиль AAA:

```
RTT(config)# aaa radius-profile RADIUS
RTT(config-aaa-radius-profile)# radius-server host 192.0.2.20
RTT(config-aaa-radius-profile)# exit
```

Укажем параметры доступа к DAS (Direct-attached storage) серверу:

```
RTT(config)# object-group network server
RTT(config-object-group-network)# ip address-range 192.0.2.20
RTT(config-object-group-network)# exit
RTT(config)# das-server CoA
RTT(config-das-server)# key ascii-text password
RTT(config-das-server)# port 3799
RTT(config-das-server)# clients object-group server
RTT(config-das-server)# exit
RTT(config)# aaa das-profile CoA
RTT(config-aaa-das-profile)# das-server CoA
RTT(config-aaa-das-profile)# exit
```

До аутентификации весь трафик из зоны trusted блокируется, в том числе DHCP и DNS запросы. Необходимо настроить разрешающие правила для пропуска DHCP и DNS запросов:

```
RTT(config)# ip access-list extended DHCP
RTT(config-acl)# rule 10
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol udp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port 68
RTT(config-acl-rule)# match destination-port 67
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# rule 11
```



```
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol udp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port any
RTT(config-acl-rule)# match destination-port 53
RTT(config-acl-rule)# enable
RTT(config-acl-rule)#exit
RTT(config-acl)# exit
```

Далее создаем правила для редиректа на портал и пропуска трафика в Интернет:

```
RTT(config)# ip access-list extended WELCOME
RTT(config-acl)# rule 10
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
RTT (config)# ip access-list extended INTERNET
RTT(config-acl)# rule 10
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# exit
```

Зададим web-ресурсы доступные без авторизации:

```
RTT(config)# object-group url defaultservice
RTT(config-object-group-url)# url http://rusteletech.ru
RTT(config-object-group-url)# exit
```

Списки фильтрации по URL находятся на сервере SoftWLC (меняется только IP-адрес сервера SoftWLC, если используется адресация отличная от данного примера, все остальное в URL оставить без изменения):

```
RTT(config)# subscriber-control filters-server-url
http://192.0.2.20:7070/Filters/file/
```

Сконфигурируем и включим BRAS, в качестве NAS IP указываем адрес интерфейса на стыке с SoftWLC, в данном примере – это IP-адрес интерфейса gigabitethernet 1/0/24:

```
RTT(config)# subscriber-control
RTT(config-subscriber-control)# aaa das-profile CoA
RTT(config-subscriber-control)# aaa sessions-radius-profile RADIUS
RTT(config-subscriber-control)# nas-ip-address 192.0.2.1
RTT(config-subscriber-control)# session mac-authentication
RTT(config-subscriber-control)# bypass-traffic-acl DHCP
RTT(config-subscriber-control)# default-service
RTT(config-subscriber-default-service)# class-map INTERNET
RTT(config-subscriber-default-service)# filter-name local defaultservice
RTT(config-subscriber-default-service)# filter-action permit
RTT(config-subscriber-default-service)# default-action redirect
http://192.0.2.20:8080/rustel_portal/
RTT(config-subscriber-default-service)# session-timeout 3600
RTT(config-subscriber-default-service)# exit
```

```
RTT(config-subscriber-control)# enable
RTT(config-subscriber-control)# exit
```

Далее необходимо сконфигурировать правила перехода между зонами безопасности.

```
RTT(config)# object-group service telnet
RTT(config-object-group-service)# port-range 23
RTT(config-object-group-service)# exit
RTT(config)# object-group service ssh
RTT(config-object-group-service)# port-range 22
RTT(config-object-group-service)# exit
RTT(config)# object-group service dhcp_server
RTT(config-object-group-service)# port-range 67
RTT(config-object-group-service)# exit
RTT(config)# object-group service dhcp_client
RTT(config-object-group-service)# port-range 68
RTT(config-object-group-service)# exit
RTT(config)# object-group service ntp
RTT(config-object-group-service)# port-range 123
RTT(config-object-group-service)# exit
```

Разрешим доступ в Интернет из зон trusted и dmz:

```
RTT(config)# security zone-pair trusted untrusted
RTT(config-zone-pair)# rule 10
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# security zone-pair dmz untrusted
RTT(config-zone-pair)# rule 10
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
RTT(config)# security zone-pair dmz trusted
RTT(config-zone-pair)# rule 10
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol any
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

Разрешим прохождение DHCP из trusted в dmz:

```
RTT (config)# security zone-pair trusted dmz
RTT (config-zone-pair)# rule 10
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol udp
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match source-port dhcp_client
RTT(config-zone-pair-rule)# match destination-port dhcp_server
```

```
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# exit
```

Разрешим прохождение ICMP к устройству, для работы BRAS необходимо открыть порты для веб-проксирования – TCP 3129/3128 (NetPort Discovery Port/Active API Server Port):

```
RTT(config)# object-group service bras
RTT(config-object-group-service)# port-range 3129
RTT(config-object-group-service)# port-range 3128
RTT(config-object-group-service)# exit
RTT(config)# security zone-pair trusted self
RTT(config-zone-pair)# rule 10
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol tcp
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# match source-port any
RTT(config-zone-pair-rule)# match destination-port bras
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair)# rule 20
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair-rule)# exit
RTT(config)# security zone-pair dmz self
RTT(config-zone-pair)# rule 20
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair-rule)# exit
RTT(config)# security zone-pair untrusted self
RTT(config-zone-pair)# rule 20
RTT(config-zone-pair-rule)# action permit
RTT(config-zone-pair-rule)# match protocol icmp
RTT(config-zone-pair-rule)# match source-address any
RTT(config-zone-pair-rule)# match destination-address any
RTT(config-zone-pair-rule)# enable
RTT(config-zone-pair-rule)# exit
RTT(config-zone-pair-rule)# exit
```

Активируем DHCP-Relay:

```
RTT(config)# ip dhcp-relay
```

Настроим SNAT в порт gigabitethernet 1/0/1:

```
RTT(config)# nat source
RTT(config-snat)# ruleset inet
RTT(config-snat-ruleset)# to interface gigabitethernet 1/0/1
RTT(config-snat-ruleset)# rule 10
RTT(config-snat-rule)# match source-address any
RTT(config-snat-rule)# action source-nat interface
RTT(config-snat-rule)# enable
```

7.45.3 Пример настройки BRAS без использования ПО SoftWLC

Задача: Настроить BRAS без поддержки SoftWLC.

Дано: подсеть с клиентами 10.10.0.0/16, подсеть для работы с FreeRADIUS-сервером 192.168.1.1/24

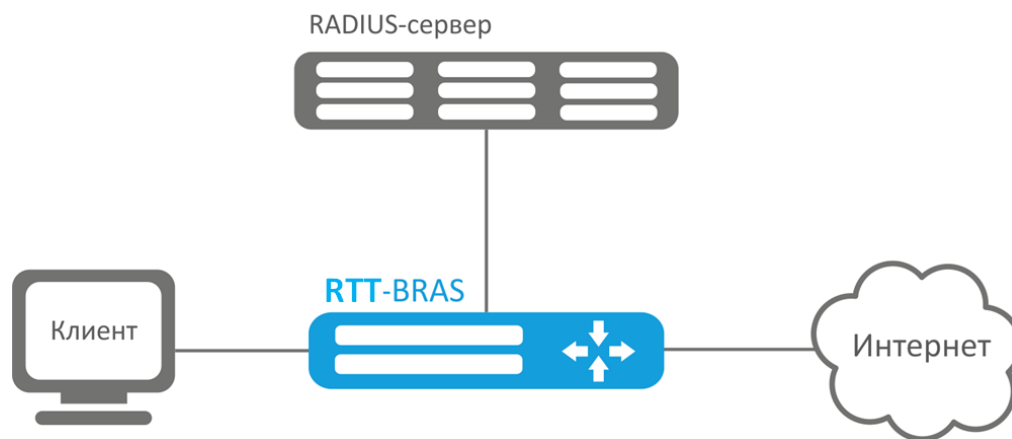


Рисунок 70 – Схема сети

Решение:

Шаг 1: Настройка RADIUS-сервера.

Для FreeRADIUS-сервера нужно задать подсеть, из которой могут приходить запросы и добавить список пользователей. Для этого в файл users в директории с файлами конфигурации FreeRADIUS сервера нужно добавить:

Профиль пользователя:

```
<MACADDR> Cleartext-Password := <MACADDR>
```

#Имя пользователя

```
User-Name = <USER_NAME>,
```

#Максимальное время жизни сессии

```
Session-Timeout = <SECONDS>,
```

#Максимальное время жизни сессии при бездействии пользователя

```
Idle-Timeout = <SECONDS>,
```

#Время на обновление статистики по сессии

```
Acct-Interim-Interval = <SECONDS>,
```

#Имя сервиса для сессии (A - сервис включен, N - сервис выключен)

```
Cisco-Account-Info = "{A|N}<SERVICE_NAME>"
```

Профиль сервиса:

```
<SERVICE_NAME> Cleartext-Password := <MACADDR>
```

Соответствует имени class-map в настройках RTT

```
Cisco-AVPair = "subscriber:traffic-class=<CLASS_MAP>",
```

Действие, которое применяет RTT к трафику (permit, deny, redirect)

```
Cisco-AVPair = "subscriber:filter-default-action=<ACTION>",
```

Возможность прохождения IP потоков (enabled-uplink, enabled-downlink, enabled, disabled)

```
Cisco-AVPair = "subscriber:flow-status=<STATUS>"
```

В файл clients.conf нужно добавить подсеть, в которой находится RTT:

```
client RTT {  
  
  ipaddr = <SUBNET>  
  
  secret = <RADIUS_KEY>  
  
}
```

В нашем случае настройка RADIUS-сервера будет выглядеть так:

в файл «clients.conf» добавляем строки:

```
client BRAS {  
  ipaddr = 192.168.1.1  
  secret = password  
  
}
```

В файл «users» добавляем строки (вместо <MAC> нужно указать MAC адрес клиента):

```
"54-E1-AD-8F-37-35" Cleartext-Password := "54-E1-AD-8F-37-35"  
User-Name = "Bras_user",  
Session-Timeout = 259200,  
Idle-Timeout = 259200,  
Cisco-AVPair += "subscriber:policer-rate-in=1000",  
Cisco-AVPair += "subscriber:policer-rate-out=1000",  
Cisco-AVPair += "subscriber:policer-burst-in=188",  
Cisco-AVPair += "subscriber:policer-burst-out=188",  
Cisco-Account-Info = "AINTERNET"  
  
INTERNET Cleartext-Password := "INTERNET"  
User-Name = "INTERNET",  
Cisco-AVPair = "subscriber:traffic-class=INTERNET",  
Cisco-AVPair += "subscriber:filter-default-action=permit"
```

Шаг 2: Настройка RTT.

Для настройки функционала BRAS необходимо наличие лицензии BRAS.

```
RTT(config)# do sh licence
```

Licence information

```
-----  
  
Name:      Rustel  
  
Version: 1.0  
  
Type:      RTT-RX  
  
S/N:       NP00000000  
  
MAC:       XX:XX:XX:XX:XX:XX  
  
Features:  
  
BRAS - Broadband Remote Access Server
```

Настройка параметров для взаимодействия с RADIUS-сервером:

```
RTT(config)# radius-server host 192.168.1.2  
RTT(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF  
RTT(config-radius-server)# source-address 192.168.1.1  
RTT(config-radius-server)# exit
```

Создадим профиль AAA:

```
RTT(config)# aaa radius-profile bras_radius  
RTT(config-aaa-radius-profile)# radius-server host 192.168.1.2  
RTT(config-aaa-radius-profile)# exit  
RTT(config)# aaa radius-profile bras_radius_servers  
RTT(config-aaa-radius-profile)# radius-server host 192.168.1.2  
RTT(config-aaa-radius-profile)# exit
```

Укажем параметры к DAS-серверу:

```
RTT(config)# das-server das  
RTT(config-das-server)# key ascii-text encrypted 8CB5107EA7005AFF  
RTT(config-das-server)# exit  
RTT(config)# aaa das-profile bras_das  
RTT(config-aaa-das-profile)# das-server das  
RTT(config-aaa-das-profile)# exit  
  
RTT(config)# vlan 10  
RTT(config-vlan)# exit
```

Далее создаем правила для редиректа на портал и пропуска трафика в Интернет:

```
RTT(config)# ip access-list extended BYPASS  
RTT(config-acl)# rule 1  
RTT(config-acl-rule)# action permit  
RTT(config-acl-rule)# match protocol udp  
RTT(config-acl-rule)# match source-address any  
RTT(config-acl-rule)# match destination-address any  
RTT(config-acl-rule)# match source-port 68  
RTT(config-acl-rule)# match destination-port 67  
RTT(config-acl-rule)# enable  
RTT(config-acl-rule)# exit  
RTT(config-acl)# rule 2  
RTT(config-acl-rule)# action permit  
RTT(config-acl-rule)# match protocol udp  
RTT(config-acl-rule)# match source-address any  
RTT(config-acl-rule)# match destination-address any  
RTT(config-acl-rule)# match source-port any  
RTT(config-acl-rule)# match destination-port 53
```

```
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config)# ip access-list extended INTERNET
RTT(config-acl)# rule 1
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol any
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config)# ip access-list extended WELCOME
RTT(config-acl)# rule 10
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol tcp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port any
RTT(config-acl-rule)# match destination-port 443
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# rule 20
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol tcp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port any
RTT(config-acl-rule)# match destination-port 8443
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# rule 30
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol tcp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port any
RTT(config-acl-rule)# match destination-port 80
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
RTT(config-acl)# rule 40
RTT(config-acl-rule)# action permit
RTT(config-acl-rule)# match protocol tcp
RTT(config-acl-rule)# match source-address any
RTT(config-acl-rule)# match destination-address any
RTT(config-acl-rule)# match source-port any
RTT(config-acl-rule)# match destination-port 8080
RTT(config-acl-rule)# enable
RTT(config-acl-rule)# exit
```

! Настройка действие фильтрации по URL обязательно, а именно, необходимо настроить фильтрацию http-проxy на BRAS для неавторизованных пользователей:

```
RTT(config)# object-group url defaultserv
RTT(config-object-group-url)# url http://rusteletech.ru
RTT(config-object-group-url)# url http://ya.ru
RTT(config-object-group-url)# url https://ya.ru
RTT(config-object-group-url)# exit
```

Сконфигурируем и включим BRAS, в качестве NAS IP указываем адрес интерфейса на стыке с RADIUS-сервером в данном примере – это IP-адрес интерфейса gigabitethernet 1/0/2:

```
RTT(config)# subscriber-control
```

```
RTT(config-subscriber-control)# aaa das-profile bras_das
RTT(config-subscriber-control)# aaa sessions-radius-profile bras_radius
RTT(config-subscriber-control)# aaa services-radius-profile bras_radius_servers
RTT(config-subscriber-control)# nas-ip-address 192.168.1.1
RTT(config-subscriber-control)# session mac-authentication
RTT(config-subscriber-control)# bypass-traffic-acl BYPASS
RTT(config-subscriber-control)# default-service
RTT(config-subscriber-default-service)# class-map BYPASS
RTT(config-subscriber-default-service)# filter-name local defaultserv
RTT(config-subscriber-default-service)# filter-action permit
RTT(config-subscriber-default-service)# default-action redirect http://192.168.1.2:8080/rustel_portal
RTT(config-subscriber-default-service)# session-timeout 121
RTT(config-subscriber-default-service)# exit
RTT(config-subscriber-control)# enable
RTT(config-subscriber-control)# exit
```

На интерфейсах, для которых требуется работа BRAS настроить (для успешного запуска требуется как минимум один интерфейс):

```
RTT(config)# bridge 10
RTT(config-bridge)# vlan 10
RTT(config-bridge)# ip firewall disable
RTT(config-bridge)# ip address 10.10.0.1/16
RTT(config-bridge)# ip helper-address 192.168.1.2
RTT(config-bridge)# service-subscriber-control any
RTT(config-bridge)# location USER
RTT(config-bridge)# protected-ports
RTT(config-bridge)# protected-ports exclude vlan
RTT(config-bridge)# enable
RTT(config-bridge)# exit
```

Сконфигурируем порт в сторону RADIUS-сервера:

```
RTT(config)# interface gigabitethernet 1/0/2
RTT(config-if-gi)# ip firewall disable
RTT(config-if-gi)# ip address 192.168.1.1/24
RTT(config-if-gi)# exit
```

Порт в сторону Клиента:

```
RTT(config)# interface gigabitethernet 1/0/3.10
RTT(config-subif)# bridge-group 10
RTT(config-subif)# ip firewall disable
RTT(config-subif)# exit
```

Настройка SNAT в порт gigabitethernet 1/0/2:

```
RTT(config)# nat source
RTT(config-snat)# ruleset factory
RTT(config-snat-ruleset)# to interface gigabitethernet 1/0/2
RTT(config-snat-ruleset)# rule 10
RTT(config-snat-rule)# description "replace 'source ip' by outgoing interface ip address"
RTT(config-snat-rule)# match protocol any
RTT(config-snat-rule)# match source-address any
RTT(config-snat-rule)# match destination-address any
RTT(config-snat-rule)# action source-nat interface
RTT(config-snat-rule)# enable
RTT(config-snat-rule)# exit
RTT(config-snat-ruleset)# exit
RTT(config-snat)# exit
```



```
RTT(config)# ip route 0.0.0.0/0 192.168.1.2
```

Изменения конфигурации вступят в действие после применения:

```
RTT(config) # do commit
RTT(config) # do confirm
```

Для просмотра информации и статистики по сессиям контроля пользователей - можно воспользоваться командой:

```
RTT # sh subscriber-control sessions status
```

Session id	User name	IP address	MAC address	Interface	Domain
1729382256910270473	Bras_user	10.10.0.3	54:e1:ad:8f:37:35	gi1/0/3.10	--

7.46 Настройка VoIP

VoIP (англ. Voice over IP) — набор протоколов, которые позволяют передавать речевую информацию посредством IP-сетей. В рамках данного устройства VoIP используется для подключения аналоговых телефонных аппаратов к IP-сети с возможностью совершения телефонных вызовов.

7.46.1 Процесс настройки SIP-профиля

Шаг	Описание	Команда	Ключи
1	Настройка SIP-профиля	RTT(config)# sip profile <NUM>	<NUM> – номер SIP-профиля, задается в виде цифры от 1 до 5.
2	Настройка основного SIP проху-сервера и сервера регистрации	RTT(config-sip-profile)# proxy primary	
3	Настройка адреса SIP проху-сервера	RTT(config-voip-sip-proxy)# ip address proxy-server <IP>	<IP> – IP-адрес проху-сервера
4	Настройка порта SIP проху-сервера	RTT(config-voip-sip-proxy)# ip port proxy-server <PORT>	<PORT> – номер UDP-порта проху-сервера, принимает значения [1..65535]. Если используется стандартный порт 5060, то его можно не указывать.
5	Настройка адреса сервера регистрации	RTT(config-voip-sip-proxy)# ip address registration-server <IP>	<IP> – IP-адрес сервера регистрации.
6	Настройка порта сервера регистрации:	RTT(config-voip-sip-proxy)# ip portregistration-server <PORT>	<PORT> – номер UDP-порта сервера регистрации, принимает значения [1..65535]. Если используется стандартный порт 5060, то его можно не указывать.
7	Активация регистрации	RTT(config-voip-sip-proxy)# registration	
8	Активация проху-сервера и сервера регистрации:	RTT(config-voip-sip-proxy)# enable	
9	Настройка адреса сервера регистрации	RTT(config-voip-sip-proxy)# ip address registration-server <IP>	<IP> – IP-адрес сервера регистрации.
10	Настройка порта сервера регистрации:	RTT(config-voip-sip-proxy)# ip portregistration-server	<PORT> – номер UDP-порта сервера регистрации, принимает значения

		<PORT>	[1..65535]. Если используется стандартный порт 5060, то его можно не указывать.
11	Указание SIP-домена, в котором находится устройство	RTT(config-sip-profile)# sip-domain address <ADDRESS>	<ADDRESS> – SIP-домен, в котором находится устройство, задается ipv4-адресом или доменным именем.
12	Активация использования SIP domain при регистрации	RTT(config-sip-profile)# sip-domain registration enable	
13	Настройка SIP-профиля	RTT(config)# sip profile <NUM>	<NUM> – номер SIP-профиля, задается в виде цифры от 1 до 5.
14	Назначение плана нумерации текущему SIP-профилю	RTT(config-sip-profile)# dialplan pattern <DNAME>	<DNAME> – имя план нумерации, задается строкой до 31 символа.
15	Активация SIP-профиля	RTT(config-sip-profile)# enable	

7.46.2 Процесс настройки FXS/FXO-портов

Шаг	Описание	Команда	Ключи
1	Переход в режим конфигурирования FXO/FXS-портов	RTT(config)# interface voice-port <NUM>	<NUM> - номер порта, принимает значение от 1 до 4.
2	Назначение абонентского номера, закрепленного за телефонным портом	RTT(config-voice-port-fxs)# sip user phone <PHONE>	<PHONE> - абонентский номер, закрепленный за телефонным портом, задается до 50 символов.
3	Назначение имя пользователя, сопоставленное с портом	RTT(config-voice-port-fxs)# sip user display-name <LOGIN>	<LOGIN> - имя пользователя, которое отображается в поле Display-Name, задается строкой до 31 символа.
4	Выбор SIP-профиля для конкретного порта.	RTT(config-voice-port-fxs)# profile sip <PROFILE>	<PROFILE> - номер SIP-профиля, задается от 1 до 5.
5	Настройка логина для аутентификации	RTT(config-voice-port-fxs)# authentication name <LOGIN>	<LOGIN> - логин для аутентификации, задается строкой до 31 символа
6	Настройка пароля для аутентификации	RTT(config-voice-port-fxs)# authentication password <PASS>	<PASS> - пароль для аутентификации, задается строкой до 16 символов.
7	Активация FXO-порта	RTT(config)# interface voice-port <NUM>	<NUM> - номер FXO-порта принимает значение от 1 до 4.
8	Назначение абонентского номера, закрепленного за телефонным портом	RTT(config-voice-port-fxo)# sip user phone <PHONE>	<PHONE> - абонентский номер, закрепленный за телефонным портом.
9	Указание UDP-порта, с которого и на который FXO-комплект будет отправлять и принимать SIP-сообщения	RTT(config-voice-port-fxo)# sip port <PORT>	<PORT> - номер UDP-порта.
10	Назначение имени пользователя, сопоставленное с портом	RTT(config-voice-port-fxo)# sip user display-name <LOGIN>	<LOGIN> - имя пользователя, которое отображается в поле Display-Name, задается строкой до 31 символа.
11	Настройка логина для аутентификации	RTT(config-voice-port-fxo)# authentication name <LOGIN>	<LOGIN> - логин для аутентификации, задается строкой до 31 символа.
12	Настройка пароля для	RTT(config-voice-port-fxo)# authentication password <PASS>	<PASS> - пароль для аутентификации,

	аутентификации	<code>fxo)# authentication password <PASS></code>	задается строкой до 16 символов.
13	Разрешение передачи номера в ТФОП	<code>RTT(config-voice-port-fxo)# pstn transmit-number</code>	
14	Запретить передачу префикса	<code>RTT(config-voice-port-fxo)# no pstn transmit-prefix</code>	
15	Активация услуги «Hotline PSTN to IP»	<code>RTT(config-voice-port-fxo)# hotline ipt</code>	
16	Номер абонента, который будет получать вызовы с ТФОП	<code>RTT(config-voice-port-fxo)# hotline number ipt <PHONE></code>	<PHONE> – номер телефона, на который осуществляется вызов при использовании услуги, принимает значение от 1 до 50. «Горячая/Теплая линия» в направлении из аналоговой телефонной линии в VoIP.

7.46.3 Процесс настройки плана нумерации

Шаг	Описание	Команда	Ключи
1	Создание плана нумерации	<code>RTT(config)# dialplan pattern <DNAME></code>	<DNAME> - имя плана нумерации, задается строкой до 31 символа.
2	Добавление правил нумерации	<code>RTT(config-dial-ruleset)# pattern <REGEXP></code>	<REGEXP> - регулярное выражение, задающее план нумерации. Задаётся строкой до 1024 символов. Правила составления регулярных выражений описаны в разделе 7.46.5
3	Активация плана нумерации	<code>RTT(config-dial-ruleset)# enable</code>	

7.46.4 Пример настройки VoIP

Задача:

Подключить аналоговые телефонные аппараты и факс-модемы к IP-сети посредством маршрутизатора RTT. В качестве проху-сервера и сервера регистрации выступает SIP-сервер, находящийся на RTT.

Решение:



Рисунок 71 – Схема сети

Настройка SIP-профиля:

```
RTT(config)# sip profile 1
```

Настройка основного SIP проху-сервера и сервера регистрации:

```
RTT(config-sip-profile) # proxy primary
```

Настройка адреса SIP проху-сервера (в качестве SIP проху-сервера использовать встроенный SIP-сервер):

```
RTT(config-voip-sip-proxy) # ip address proxy-server 192.0.2.5
```

Настройка порта SIP проху-сервера:

```
RTT(config-voip-sip-proxy) # ip port proxy-server 5080
```

Если используется стандартный порт 5060, то его можно не указывать.

Если необходимо использовать регистрацию, то необходимо выполнить следующие пункты:

Настройка адреса сервера регистрации (в качестве сервера регистрации использовать встроенный SIP-сервер):

```
RTT(config-voip-sip-proxy) # ip address registration-server 192.0.2.5
```

Настройка порта сервера регистрации:

```
RTT(config-voip-sip-proxy) # ip port registration-server 5080
```

Если используется стандартный порт 5060, то его можно не указывать.

Активация регистрации:

```
RTT(config-voip-sip-proxy) # registration
```

Активация проху-сервера и сервера регистрации:

```
RTT(config-voip-sip-proxy) # enable
```

На этом конфигурация SIP проху-сервера и сервера регистрации закончена:

```
RTT(config-voip-sip-proxy) # exit
```

Далее продолжается настройка SIP-профиля.

Настройка SIP-домена:

```
RTT(config-sip-profile) # sip-domain address sipdomain.com
```

Если необходимо использовать SIP Domain для регистрации:

```
RTT(config-sip-profile) # sip-domain registration enable
```

В такой конфигурации все вызовы будут направлены SIP проху-серверу. Если необходимо указать другое направление для исходящих вызовов, то необходимо проделать следующее:

Создать план нумерации, см. раздел 7.46.5.

Далее созданный план маршрутизации, необходимо присвоить SIP-профилю:

```
RTT(config) # sip profile 1
```

```
RTT(config-sip-profile) # dialplan pattern firstDialplan
```

На этом настройка плана нумерации для SIP-профиля закончена.

Активация SIP-профиля:

```
RTT(config-sip-profile) # enable
```

На этом минимально необходимая настройка SIP-профиля закончена:

```
RTT(config-sip-profile) # exit
```

Следующим этапом является настройка абонентских портов:

```
RTT(config) # interface voice-port 1
```

Указать абонентский номер:

```
RTT(config-voice-port-fxs) # sip user phone 4101
```

Указать отображаемое имя:

```
RTT(config-voice-port-fxs) # sip user display-name user-one
```

Используемый SIP-профиль:

```
RTT(config-voice-port-fxs) # profile sip 1
```

Настройка логина и пароля для аутентификации:

```
RTT(config-voice-port-fxs) # authentication name login-4101
```

```
RTT(config-voice-port-fxs) # authentication password superpassword
```

На этом минимальная настройка абонентского порта закончена:

```
RTT(config-voice-port-fxs) # exit
```

7.46.5 Пример настройки плана нумерации

Задача:

Настроить план нумерации так, чтобы вызовы на локальных коммутировались локально, а вызовы на все остальные направления через SIP-proxy.

Решение:

Создать план нумерации:

```
RTT(config)# dialplan pattern firstDialplan
```

План нумерации задается при помощи регулярных выражений:

```
RTT(config-dial-ruleset)# pattern "<regular expressions>"
```

Для задачи, обозначенной выше "<regular expressions>" будет иметь вид:

"S5, L5 (410[1-3]@{local} | [xABCD*#].S)"

где:

410[1-3]@{local} – вызовы на номера 4101, 4102, 4103 будут коммутироваться локально;

[xABCD*#].S – вызовы на все остальные номера будут направлены к SIP-прокси.

Активировать план нумерации:

```
RTT(config-dial-ruleset)# enable
```

Настройка плана нумерации закончена.

```
RTT(config-dial-ruleset)# exit
```

Структура регулярного выражения:

Sxx, Lxx (),

где:

xx – произвольные значения таймеров S и L;

() – границы плана нумерации.

Основой являются обозначения для записи последовательности набранных цифр. Последовательность цифр записывается с помощью нескольких обозначений: цифры, набираемые с клавиатуры телефона: 0, 1, 2, 3, ..., 9, # и *.



Использование символа # в плане нумерации может блокировать завершение набора с помощью этой клавиши!

Последовательность цифр, заключённая в квадратные скобки, соответствует любому из заключённых в скобки символу.

- Пример: ([1239]) – соответствует любой из цифр 1, 2, 3 или 9.

Через тире может быть указан диапазон символов. Чаще всего используется внутри квадратных скобок.

- Пример 1: (1-5) – любая цифра от 1 до 5.
- Пример 2: ([1-39]) – пример из предыдущего пункта с иной формой записи.

Символ X соответствует любой цифре от 0 до 9.

- Пример: (1XX) – любой трёхзначный номер, начинающийся на 1.

«.» – повторение предыдущего символа от 0 до бесконечности раз.

«+» – повторение предыдущего символа от 1 до бесконечности раз.

{a,b} – повторение предыдущего символа от a до b раз;

{a,} – повторение предыдущего символа не меньше a раз;

{,b} – повторение предыдущего символа не больше b раз.

- Пример: (810X.) – международный номер с любым количеством цифр.

Настройки, влияющие на обработку плана нумерации:

- Interdigit Long Timer (буква «L» в записи плана нумерации) – время ожидания ввода следующей цифры в том случае, если нет шаблонов, подходящих под набранную комбинацию;
- Interdigit Short Timer (буква «S» в записи плана нумерации) – время ожидания ввода следующей цифры, если с набранной комбинацией полностью совпадает хотя бы один шаблон и при этом имеется еще хотя бы один шаблон, до полного совпадения с которым необходимо осуществить донабор номера.

Дополнительные возможности:

1. Замена набранной последовательности

Синтаксис: `<arg1:arg2>`

Данная возможность позволяет заменить набранную последовательность на любую последовательность набираемых символов. При этом второй аргумент должен быть указан определённым значением, оба аргумента могут быть пустыми.

- Пример: (<83812:> XXXXXX) – данная запись будет соответствовать набранным цифрам 83812, но эта последовательность будет опущена и не будет передана на SIP-сервер.

2. Вставка тона в набор

При выходе на межгород (в офисных станциях – на город) привычно слышать ответ станции, что можно реализовать вставкой запятой в нужную позицию последовательности цифр.

- Пример: (8, 770) – при наборе номера 8770 после цифры 8 будет выдан непрерывный тон.

3. Запрет набора номера

Если в конце шаблона номера добавить восклицательный знак '!', то набор номеров, соответствующих шаблону, будет заблокирован.

- Пример: (8 10X xxxxxxx ! | 8 xxx xxxxxxx) – выражение разрешает набор только междугородних номеров и исключает международные вызовы.

4. Замена значений таймеров набора номера

Значения таймеров могут быть назначены как для всего плана нумерации, так и для определённого шаблона. Буква «S» отвечает за установку «*Interdigit Short Timer*», а «L» – за «*Interdigit Long Timer*». Значения таймеров может быть указано для всех шаблонов в плане нумерации, если значения перечислены до открывающейся круглой скобки.

- Пример: S4 (8XXX.) или S4,L8 (XXX)

Если эти значения указаны только в одной из последовательностей, то действуют только для неё. Также в этом случае не надо ставить двоеточие между ключом и значением таймаута, значение может быть расположено в любом месте шаблона.

- Пример: (S4 8XXX. | XXX) или ([1-5] XX S0) – запись вызовет мгновенную передачу вызова при наборе трехзначного номера, начинающегося на 1,2, ... , 5.

5. Набор по прямому адресу (IP Dialing)

Символ «@», поставленный после номера, означает, что далее будет указан адрес сервера, на который будет отправлен вызов на набранный номер. Рекомендуется использовать «*IP Dialing*», а также приём и передачу вызовов без регистрации («*Call Without Reg*», «*Answer Without Reg*»). Это может помочь в случае отказа сервера.

Кроме того, формат адреса с IP Dialing может быть использован в номерах, предназначенных для переадресации звонков.

- Пример 1: (8 xxx xxxxxxx) – 11-значный номер, начинающийся на 8.
- Пример 2: (8 xxx xxxxxxx | <:8495> xxxxxxx) – 11-значный номер, начинающийся на 8, если введён семизначный, то добавить к передаваемому номеру 8495.
- Пример 3: (0[123] | 8 [2-9]xx [2-9]xxxxxx) – набор номеров экстренных служб, а так же некоторого странного набора междугородних номеров.
- Пример 4: (S0 <:82125551234>) – быстрый набор указанного номера, аналог режима «Hotline» на других шлюзах.
- Пример 5: (S5 <:1000> | xxxx) – данный план нумерации позволяет набрать любой номер, состоящий из цифр, а если ничего не введено в течение 5 секунд, вызвать номер 1000 (допустим, это секретарь).
- Пример 6: (8, 10x. | 1xx@10.110.60.51:5060) - данный план нумерации позволяет набирать номера начинающиеся на 810 и содержащие как минимум одну цифру после "810". После ввода 8 будет выдан сигнал "ответ станции". А также набор трехзначных номеров, начинающихся на "1", Invite на которые будет отправлен на IP-адрес 10.110.60.51 и порт 5060.
- Пример 7: (S3 *xx# | #xx# | #xx# | *xx*x#) - управление и использование ДВО.

Иногда может потребоваться совершать звонки локально внутри устройства. При этом, если IP-адрес устройства не известен или периодически изменяется, удобно использовать

в качестве адреса сервера зарезервированное слово «{local}», что означает отправку соответствующей последовательности цифр на собственный адрес устройства.

- Пример: (123@{local}) – вызов на номер 123 будет обработан локально внутри устройства.

8 ЧАСТО ЗАДАВАЕМЫЕ ВОПРОСЫ



Изменения конфигурации вступают в действие после применения:

```
RTT# commit
Configuration has been successfully committed
RTT# confirm
Configuration has been successfully confirmed
```

- Не удалось получить маршруты по BGP и/или OSPF, сконфигурированных в VRF. Соседство успешно устанавливается, но в записи маршрутов в RIB отказано:
%ROUTING-W-KERNEL: Can not install route. Reached the maximum number of BGP routes in the RIB

Необходимо выделить ресурс RIB для VRF, по умолчанию он равен нулю. Делаем это в режиме конфигурирования VRF:

```
RTT(config)# ip vrf <NAME>
RTT(config-vrf)# ip protocols ospf max-routes 12000
RTT(config-vrf)# ip protocols bgp max-routes 1200000
RTT(config-vrf)# end
```

- Закрываются сессии SSH/Telnet проходящие через маршрутизатор RTT.

Для поддержания сессии активной необходимо настроить передачу keepalive пакетов. Опция отправки keepalive настраивается в клиенте SSH, например, для клиента PuTTY раздел "Соединение".

В свою очередь, на маршрутизаторе можно выставить время ожидания до закрытия неактивных сессий TCP (в примере выставлен 1 час):

```
RTT(config)# ip firewall sessions tcp-established-timeout 3600
```

- На интерфейсе был отключен firewall (ip firewall disable), после внесения этого интерфейса в security zone, удаления из конфигурации ip firewall disable и применения изменений - доступ для активных сессий с данного порта не закрылся, согласно правилам security zone-pair.

Изменения в конфигурации Firewall будут действовать только для новых сессий, сброса активных сессий в Firewall не происходит. Отчистить активные сессии в firewall можно командой:

```
RTT# clear ip firewall session
```

- Не поднимается LACP на портах XG RTT-R800

По умолчанию на port-channel режим speed 1000M, необходимо выставить speed 10G.

```
RTT(config)# interface port-channel 1
RTT(config-port-channel)# speed 10G
```

- Как полностью очистить конфигурация RTT, и как сбросить на заводскую конфигурацию?

Очистка конфигурации происходит путем копирования пустой конфигурации в candidate-config и применения его в running-config.

```
RTT# copy system:default-config system:candidate-config
```

Процесс сброса на заводскую конфигурацию аналогичен.

```
RTT# copy system:factory-config system:candidate-config
```

- **Как привязать subinterface к созданным VLAN?**

При создании саб-интерфейса, VLAN создается и привязывается автоматически (прямая зависимость индекс sub - VID).

```
RTT(config)# interface gigabitethernet 1/0/1.100
```

После применения можно наблюдать информационные сообщения:

```
2016-07-14T012:46:24+00:00 %VLAN: creating VLAN 100
```

- **Есть ли функционал в маршрутизаторах серии RTT-R для анализа трафика?**

В маршрутизаторах серии RTT-R реализована возможность анализировать трафик на интерфейсах из CLI. Сниффер запускается командой monitor.

```
RTT# monitor gigabitethernet 1/0/1
```

- **Как настроить ip prefix-list 0.0.0.0/0?**

Ниже приведен пример конфигурации префикс листа, разрешающего прием маршрута по умолчанию.

```
RTT(config)# ip prefix-list Rustel
RTT(config-pl)# permit default-route
```

- **Проблема прохождения асинхронного трафика.**

В случае организации сети с асинхронной маршрутизацией Firewall будет запрещать "неправильный (ошибочный)" входящий трафик (не открывающий новое соединение и не принадлежащий никакому установленному соединению) из соображений безопасности.

Разрешающее правило в Firewall не решит поставленную задачу для подобных схем.

Решить задачу можно, отключив Firewall на входном интерфейсе:

```
RTT(config-if-gi)# ip firewall disable
```

ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Для получения технической консультации по вопросам эксплуатации оборудования обращайтесь в отдел технической поддержки производителя по адресу:

Российская Федерация, 115419, г. Москва, ул. Орджоникидзе, д. 11 строение 40

Телефон/факс: +7 (495)234-9-777

E-mail: support@rusteletech.ru

Почтовый адрес: 115419, г. Москва, а/я 12.

Официальный сайт производителя: rusteletech.ru