Маршрутизатор (RedBox)

TOPAZ FW

РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ

ПЛСТ.465277.305 РЭ

Москва 2024

ОГЛАВЛЕНИЕ

1 ОПИСАНИЕ И РАБОТА 4
1.1 Назначение изделия 4
1.2 Модификации и условные обозначения 4
1.3 Технические характеристики 7
1.3.1 Конструкция 7
1.3.2 Рабочие условия эксплуатации 7
1.3.3 Безопасность и электромагнитная совместимость 7
1.3.4 Надежность 7
1.3.5 Питание 8
1.3.6 Характеристики устройства 8
1.3.7 Синхронизация времени 8
1.3.8 Интерфейсы передачи данных 9
1.4 Комплектность 10
1.5 Устройство и работа 10
1.5.1 Функциональные возможности 10
1.5.2 Встроенная система безопасности 11
1.5.3 Работа кнопок и индикаторов 12
1.6 Конфигурирование устройства 12
1.6.1 Подключение к командной строке 12
1.6.2 Команды и утилиты для работы с устройством 14
1.7 VRF 20
Создание интерфейсов 21
1.7.1 GRE 21
1.7.2 Loopback 22
1.7.3 VLAN 23
1.7.4 Bridge 23
1.8 Подсистема router 24
1.8.1 FRR 24
1.8.2 Basic Commands 25
1.8.3 Расширенное ведение журнала 39
1.8.4 BGP 41
1.8.5 LDP 135
1.8.6 OSPF 140
1.8.7 VRRP 164
1.9 Настройка правил межсетевого экрана 166
1.9.1 Правила и действия 166

1.9.2 Таблицы ipatables....167
1.9.3 Утилита iptables....168
1.10 Настройка функций безопасности....171
1.10.1 Конфигурирование порта управления....171
1.10.2 Подсистема регистрации событий безопасности....171
1.10.3 Подсистема проверки целостности....173
1.10.4 Подсистема криптозащиты каналов связи....179
1.10.5 Подсистема аудита....179
1.11 Работа в режиме RedBox....192
2 МАРКИРОВКА И ПЛОМБИРОВАНИЕ....192
3 УПАКОВКА....193
4 ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ....193
5 ТРАНСПОРТИРОВАНИЕ И ХРАНЕНИЕ....193
6 УТИЛИЗАЦИЯ....194
7 ИСПОЛЬЗОВАНИЕ ПО НАЗНАЧЕНИЮ....194
7.1 Эксплуатационные ограничения и меры безопасности....194
7.2 Монтаж....194
7.2.1 Подготовка к монтажу....194
7.2.2 Установка на DIN-рейку....195
7.2.3 Внешние подключения....195
7.2.4 Шина T-BUS....195
7.2.5 Подключение питания....196
7.2.6 Монтаж устройства TOPAZ FW MX710....198
7.2.7 Подключение к сети Ethernet....203
7.2.8 Горячая замена блока питания в модификации TOPAZ FW MX710....206
ПРИЛОЖЕНИЕ А (Назначение контактов и портов)....207
Таблица А.1 – Обозначения клемм и портов TOPAZ FW MX240, TOPAZ FW MX681....207
Таблица А.2 – Назначение клемм и портов TOPAZ FW MX710....207
ПРИЛОЖЕНИЕ Б (Назначение кнопок и индикаторов)....208
Таблица Б.1 – Обозначения кнопок и индикаторов....208
Таблица Б.2 – Назначение светодиодных индикаторов TOPAZ FW MX710....208
ПРИЛОЖЕНИЕ В (Внешний вид устройства)....209
ПРИЛОЖЕНИЕ Г (Подключение к устройству с помощью утилиты PuTTY)....212

Настоящее руководство по эксплуатации (РЭ) предназначено для ознакомления со сведениями о конструкции, принципе действия, технических характеристиках маршрутизатора (RedBox) **TOPAZ FW** (далее по тексту – устройство), его составных частях, указания, необходимые для правильной и безопасной эксплуатации, технического обслуживания, ремонта, хранения и транспортирования, а также схемы подключения устройства к цепям питания, телемеханики и передачи данных.

Перед началом работы с устройством необходимо ознакомиться с настоящим РЭ.

РЭ предназначено для эксплуатационного персонала и инженеров-проектировщиков АСУ ТП, систем телемеханики и диспетчеризации.

В СВЯЗИ С ПОСТОЯННОЙ РАБОТОЙ ПО СОВЕРШЕНСТВОВАНИЮ ИЗДЕЛИЯ, В КОНСТРУКЦИЮ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ МОГУТ БЫТЬ ВНЕСЕНЫ ИЗМЕНЕНИЯ, НЕ УХУДШАЮЩИЕ ЕГО ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ И НЕ ОТРАЖЕННЫЕ В НАСТОЯЩЕМ ДОКУМЕНТЕ.

1 ОПИСАНИЕ И РАБОТА

1.1 Назначение изделия

Устройство является шлюзом безопасности и предназначено для построения защищенных сетей в задачах удаленного управления и мониторинга промышленных объектов. Устройство осуществляет управление пересылкой пакетов между различными сегментами сети на основе правил и таблиц маршрутизации. Для принятия решений о пересылке пакетов используется информация о топологии сети и определённые правила, заданные администратором.

1.2 Модификации и условные обозначения

Функциональные возможности устройства, количество и тип интерфейсов передачи данных определяются типом базовой платы и количеством/типом плат расширений. Количество и тип интерфейсов передачи данных устройства, а также наличие дополнительных функциональных возможностей зависят от конкретного исполнения.

Полная расшифровка заказных обозначений устройства приведена в таблице 1.

Для формирования наименования устройства необходимо вписать на место каждой позиции соответствующий код. Квадратными скобками выделены необязательные позиции и позиции, на место которых можно вписать одновременно несколько вариантов кода.

Таблица 1 – Расшифровка кода заказа устройства

TOPAZ FW A [B1- … -Bx]-C-D-E-F-G (H-I-J)		
Позиция	**Код**	**Описание**
A	MX240 Exx [1)]	На базе платформы MX240
	MX681 Exx [1)]	На базе платформы MX681
	MX710	На базе платформы MX710
Коммуникационные порты Ethernet [2)]		
B1- … -Bx	nGSFP	Ethernet 1000 Мбит/с SFP [3)]
	nGTXSFP	Ethernet 1000 Мбит/с combo-port RJ-45/SFP [3)]
	nGTx	Ethernet 1000 Мбит/с TX RJ-45
	nTx	Ethernet 100 Мбит/с TX RJ-45
	nFxS	Ethernet 100 Мбит/с FX LC single-mode (только для MX240 и MX681)

TOPAZ FW A [B1- … -Bx]-C-D-E-F-G (H-I-J)		
Позиция	**Код**	**Описание**
	nFxM	Ethernet 100 Мбит/с FX LC multi-mode (только для MX240 и MX681)
Коммуникационные порты RS-485 [2)]		
C	nR	Порт RS-485
Конструктивное исполнение		
D	–	В пластиковом корпусе
	MR	В металлическом корпусе 1U для установки в стойку 19” (только для MX710)
Наличие твердотельного накопителя		
E	SSDm [4)]	SSD накопители, где «m» - суммарный объем ПЗУ накопителей SSD в Гб
Каналы питания		
F	–	Один вход питания 24 В, DC (рабочий диапазон напряжения 9 – 36 В)
	LV	Один вход питания 24 В, DC (рабочий диапазон напряжения 9 – 36 В)
	24/48 [5)]	Один вход питания 24/48 В, DC (рабочий диапазон напряжения 18 – 75 В)
	HV	Один вход питания 220 В, DC/AC (рабочий диапазон напряжения 90 – 265 В (AC), 100 – 365 В (DC))
	2LV	Два входа питания 24 В, DC (рабочий диапазон напряжения 9 – 36 В)
	24/48-24/48 [5)]	Два входа питания 24/48 В, DC (рабочий диапазон напряжения 18 – 75 В)
	2HV	Два независимых встроенных источника питания 220 В, DC/AC (рабочий диапазон напряжения 90 – 265 В (AC), 100 – 365 В (DC))
	LV-HV	Один вход питания 24 В, DC (рабочий диапазон напряжения 9 – 36 В) Один вход питания 220 В, DC/AC (рабочий диапазон напряжения 90 – 265 В (AC), 100 – 365 В (DC))
	24/48-HV [5)]	Один вход питания 24/48 В, DC (рабочий диапазон напряжения 18 – 75 В) Один вход питания 220 В, DC/AC (рабочий диапазон напряжения 90 – 265 В (AC), 100 – 365 В (DC))
Режим работы RedBox		
G [6)]	RB	Работа в режиме RedBox
Средства защиты сети		
H	-	Отсутствуют дополнительные средства защиты сети
	CSG [7)]	CybSec Gateway (Шлюз безопасности)
	IDS [4)]	CybSec IDS (Средство обнаружения вторжений)
Сертифицированная ОС		
I	-	Отсутствует сертифицированная ОС на базе Linux
	ОС	Сертифицированная ОС на базе Linux

TOPAZ FW A [B1- … -Bx]-C-D-E-F-G (H-I-J)		
Позиция	**Код**	**Описание**
Дополнительное ПО		
J	-	Отсутствует дополнительное ПО
	01 [7)]	TCC-Dcrypt, в комплекте с лицензиями и сертификатами
	02 [7)]	ИнфоТЕКС-Vipnet, в комплекте с лицензиями и сертификатами
	03 [7)]	КодБезопасности-Континент АП, в комплекте с лицензиями и сертификатами
	04 [4)]	НПП Гамма Кречет, в комплекте с лицензиями и сертификатами

Примечания:

1) E(xx) - общее количество (xx) портов Ethernet устройства, задается в заказном обозначении. Максимальное суммарное количество портов Ethernet – 32.
2) n – количество портов соответствующеготипа, задается в заказном обозначении.
3) SFP-модули заказываются дополнительно:
   - TOPAZ SFP-100-01-MM – 100 мегабитный многомодовый SFP-модуль
   - TOPAZ SFP-100-01-SM – 100 мегабитный одномодовый SFP-модуль
   - TOPAZ SFP-1G-10-SM – гигабитный одномодовый SFP-модуль, дальность передачи 10 км
   - TOPAZ SFP-1G-15-SM – гигабитный одномодовый SFP-модуль, дальность передачи 15 км
   - TOPAZ SFP-1G-40-SM – гигабитный одномодовый SFP-модуль, дальность передачи 40 км
   - TOPAZ SFP-1G-01-MM – гигабитный многомодовый SFP-модуль, дальность передачи 1 км
   - TOPAZ SFP-1G-02-MM – гигабитный многомодовый SFP-модуль, дальность передачи 2 км.
4) Только для платформы MX681 с использованием SSD.
5) Только для модификации MR.
6) Опционально для платформы MX681.
7) Опционально для платформ MX240 и MX681.

Пример записи обозначения устройства **TOPAZ FW** при заказе:

с двумя Ethernet 1000 Мбит/с TX RJ-45, двумя Ethernet 100 Мбит/с TX RJ-45, двумя входами питания 24 В:

«Маршрутизатор (RedBox) TOPAZ FW MX240 E4 2GTx-2Tx-2LV»

с двумя Ethernet 1000 Мбит/с TX RJ-45, шестью Ethernet 100 Мбит/с TX RJ-45, двумя входами питания 24 В:

«Маршрутизатор (RedBox) TOPAZ FW MX240 E8 2GTx-6Tx-2LV»

с двумя Ethernet 1000 Мбит/с TX RJ-45, четырьмя Ethernet 100 Мбит/с FX LC multi-mode, входом питания 220 В:

«Маршрутизатор (RedBox) TOPAZ FW MX240 E6 2GTx-4FxM-HV»

с четырьмя комбо Ethernet 100/1000 Мбит/с TX RJ-45/1000 Мбит/с SFP, восемью Ethernet 10/100 Мбит/с TX RJ-45, в металлическом корпусе для установки в стойку 19”, с двумя входами питания 220 В:

«Маршрутизатор (RedBox) TOPAZ FW MX710 4GTxSFP-8Tx- MR-2HV»

1.3 Технические характеристики

1.3.1 Конструкция

Конструктивно устройства на базе аппаратной платформы MX240 и MX681 выполнены в пластиковом корпусе, не поддерживающем горение с креплением для установки на DIN-рейку. Вентиляционные отверстия корпуса расположены сверху и снизу корпуса. Степень защиты от проникновения внутрь твердых частиц, пыли и воды – не ниже IP20 по ГОСТ 14254-2015. По устойчивости к механическим воздействиям, устройство относится к классу М40 по ГОСТ 30631-99. Габаритные размеры устройства (ШхВхГ) не более 180х99х124 мм. Масса устройства не более 1 кг.

Устройства на базе аппаратной платформы MX710 выполнены в металлическом корпусе 2U и предназначены для установки в 19” стойку. Габаритные размеры устройства (ШхВхГ) не более 430х85х370 мм

Внешний вид, описание входов, выходов и индикаторов устройства приведены в приложении А настоящего руководства.

1.3.2 Рабочие условия эксплуатации

По рабочим условиям эксплуатации (климатическим воздействиям) устройство соответствует изделиям группы С2 по ГОСТ Р 52931-2008. По устойчивости к воздействию атмосферного давления устройство соответствует группе Р2 по ГОСТ Р 52931-2008.

Таблица 2 – Рабочие условия эксплуатации

Параметр	Значение
Температура окружающего воздуха, ^{0}C	от -40 до +70
Относительная влажность воздуха при температуре 30 ^{0}C и ниже, %	до 100
Атмосферное давление воздуха, кПа	от 60 до 106,7

1.3.3 Безопасность и электромагнитная совместимость

По устойчивости к электромагнитным помехам устройство соответствует ГОСТ Р 51318.11-2006 для класса А группы 1, и ГОСТ Р 51317.6.5-2006 для оборудования, применяемого на электростанциях и подстанциях.

Радиопомехи не превышают значений, установленных для класса А по ГОСТ 30805.22-2013, для класса А по ГОСТ 30804.3.2-2013.

Устройство, в части защиты от поражения электрическим током, соответствует требованиям ГОСТ 12.2.091-2012. Класс защиты от поражения электрическим током I по ГОСТ 12.2.007.0-75.

Электрическое сопротивление изоляции устройства не менее 2,5 МОм. Электрическая прочность изоляции устройства выдерживает без разрушения испытательное напряжение 2500 В, 50 Гц в течение 1 мин.

Устройство соответствует требованиям технических регламентов Таможенного союза ТР ТС 004/2011 «О безопасности низковольтного оборудования», ТР ТС 020/2011 «Электромагнитная совместимость технических средств».

1.3.4 Надежность

Устройство является восстанавливаемым, ремонтируемым изделием, предназначенным для круглосуточной эксплуатации в стационарных условиях в производственных помещениях. Режим работы устройства непрерывный. Продолжительность непрерывной работы не ограничена. Норма средней наработки на отказ в нормальных условиях применения составляет 200 000 ч. Полный средний срок службы составляет 30 лет. Среднее время восстановления

работоспособности на объекте эксплуатации (без учета времени прибытия персонала и при наличии ЗИП) не более 30 минут.

1.3.5 Питание

Количество и тип каналов питания устройства зависят от исполнения по питанию. Характеристики каналов питания приведены в таблице ниже.

При наличии двух встроенных блоков питания (далее – БП) в модификации TOPAZ FW MX710 реализована функция горячей замены БП.

Таблица 3 – Характеристики питания

Наименование параметра	Значение
Количество каналов питания	до 2
Номинальное напряжение питания, В: - канал 24 В - канал 24/48 В - канал 220 В	 от 9 до 36 (DC) от 18 до 75 (DC) от 90 до 265 (AC); от 100 до 365 (DC)
Частотный диапазон напряжения питания 220 В, Гц	от 45 до 55
Ток потребления канала питания 220 В, не более, А	0,4
Потребляемая мощность плат устройства, не более, Вт	45

Кратковременные перерывы питания (до 200 мс) не влияют на работу устройства. При нарушении питания на время более 200 мс, устройство корректно завершает свою работу, а при восстановлении напряжения питания устройство переходит в рабочий режим автоматически. Под корректным завершением работы в данном случае понимается отсутствие передачи ложной информации и потери конфигурационной информации. Устройство обеспечивает нормальную работу при произвольном изменении напряжения питания в пределах рабочего диапазона. Время установления рабочего режима при восстановлении питания не более 10 с.

Конфигурация устройства сохраняется в энергонезависимой памяти, которая обеспечивает сохранение параметров, при отсутствии напряжения питания, в течение 30 лет.

1.3.6 Характеристики устройства

Технические характеристики устройства приведены в таблице ниже.

Таблица 4 – Характеристики устройства

Наименование параметра	Значение
Операционная система	ТОС TOPAZ Linux
Слот для Flash-карты [1)]	microSD
Примечания: 1) Отсутствует в MX710	

1.3.7 Синхронизация времени

Характеристики синхронизации времени приведены в таблице ниже.

Таблица 5 – Характеристики синхронизации времени

Наименование параметра	Значение
Уход локальных часов без внешнего питания, с / сутки, не более	± 1
Уход локальных часов при отсутствии синхронизации по сигналам точного времени, с / сутки, не более	± 0,5
Точность синхронизации времени:	

Наименование параметра	Значение
- по протоколам ГОСТ Р МЭК 60870-5-101/104	±2 мс
- по протоколам NTP, SNTP	±100 мкс
- по протоколу PTP	±1 мкс

1.3.8 Интерфейсы передачи данных

Количество и тип каналов передачи данных обозначается в заказной кодировке устройства.

Таблица 6 – Технические характеристики интерфейса Ethernet

Заказное обозначение	Тип разъема	Скорость передачи данных, Мбит/с
nGSFP	SFP-корзина	10/100/1000
nGTx	порт RJ-45	
nGTXSFP	комбо-порт RJ-45/SFP	
nTx	порт RJ-45	10/100
nFxS	порт LC (одномодовое оптоволокно)	
nFxM	порт LC (многомодовое оптоволокно)	

Таблица 7 – Технические характеристики оптических каналов связи Ethernet

Наименование параметра		Одномодовое оптоволокно	Многомодовое оптоволокно
Сечение, мкм		9/125	50/125; 62,5/125
Дальность передачи, км	порт LC	15	2
	SFP-модуль	Определяется установленным SFP	Определяется установленным SFP
Длина волны, нм		1310	1310
Мощность передатчика, дБм		от -20 до 0	от -23,5 до -14
Чувствительность приемника, дБм		до -32	до -31

Примечание Комбо-порт GTXSFP работает в режиме автоматического переключения. При одновременном подключении ко входу RJ-45 и SFP, активен только вход SFP.

Примечание Скорость передачи данных порта SFP соответствует скорости передачи данных SFP-модуля

Таблица 8 – Поддерживаемые технологии Ethernet

Технологии	Описание
Поддерживаемые стандарты	IEEE 802.3 10BaseT; IEEE 802.3u 100BASE-TX, 100BASE-FX; IEEE 802.3z 1000BASE-X; IEEE 802.3ab 1000BASE-T; IEEE 802.3x управление потоком; IEEE 802.3az Ethernet с энергосберегающим режимом IEEE 802.1D-2004 STP, QoS; IEEE 802.1d STP; IEEE 802.1w RSTP; IEEE 802.1Q тегирование трафика.
Промышленные протоколы	Ethernet/IP; ГОСТ Р МЭК 60870-5-104; Modbus/TCP; IEC 61850

Технологии	Описание
Управление	SSH; Console – CLI; Web.
Протоколы фильтрации трафика	VLAN на основе портов
Протоколы резервирования сети	STP/RSTP; PRP; HSR
Информационная безопасность	Authentication Certificate - SSL Certificate/SSH Key Regenerate; 802.1X – Port Based; Port Security – Static MAC Port Lock.
Протоколы синхронизации времени	ГОСТ Р МЭК 60870-5-104; NTP Server/Client; IEEE 1588v2 (PTP v2)

1.4 Комплектность

Комплект поставки указывается в индивидуальном паспорте устройства.

В стандартный комплект поставки входят:

1) маршрутизатор (RedBox) TOPAZ FW;
2) паспорт;
3) штекер MC 1,5/5-ST-3,81;
4) шинные соединители ME 22.5 TBUS 1.5/5-ST-3,81;*
5) разъем MSTBT 2,5/4-ST.*

Примечание: * – для MX240 и MX681. Количество шинных соединителей и клеммных блоков согласно индивидуальному паспорту устройства;

Эксплуатационная документация доступна на сайте: http://www.tpz.ru

1.5 Устройство и работа

1.5.1 Функциональные возможности

Устройство является высокопроизводительным многоцелевым сетевым маршрутизатором, объединяющим в себе традиционные сетевые функции и комплексный многоуровневый подход к безопасности маршрутизации, что позволяет обеспечить надежную защиту для промышленной и корпоративной среды. Устройство реализует функции межсетевого экрана с использованием новейших средств обеспечения безопасности данных, шифрования, аутентификации и защиты от вторжений.

Устройство работает под управлением технологической операционной системы TOPAZ Linux и реализует следующие базовые функции:

- прием информации по цифровым каналам связи;
- выполнение прикладных программ;
- автоматическое накопление, хранение и передача информации по цифровым каналам связи;
- ведение системного времени и его автоматическая коррекция/синхронизация по сигналам точного времени.
- самодиагностика и тестирование работоспособности первичных преобразователей (датчиков);
- ведение журнала событий.
- синхронизации собственных часов от внешней сети по протоколам PTP, NTP и SNTP;
- синхронизации времени подконтрольных устройств.

В «Журнале событий» устройства автоматически фиксируются время и даты наступления следующих событий:

- попыток несанкционированного доступа;

- фактов изменения данных;
- перезапуск устройства;
- фактов корректировки времени с обязательной фиксацией времени до и после коррекции или величины коррекции времени, на которую было скорректировано устройство;
- результатов самодиагностики;
- отключения питания.

Таблица 9 – Функциональные возможности маршрутизатора

Технологии	Описание
Межсетевой экран	статическая фильтрация пакетов; фильтрация по IP адресам, протоколам и портам; трансляция адресов источника и назначения NAT; отслеживание TCP сессий, контроль корректности установления соединения; поддержка фильтрации с использованием адресных листов.
Маршрутизация	статическая маршрутизация; маршрутизация на основе политик; маршрутизация на основе правил для интерфейсов; динамическая маршрутизация: RIP v1/v2, OSPFv2, BGPv4; резервирование маршрутизаторов по протоколу VRRP.
Коммутация	поддержка STP/RSTP/MSTP поддержка VLAN IEEE802.1q
VPN	IPsec с использованием туннельного и транспортного режимов, авторизация по сертификату или PSK ключу, протоколы AH/ESP аппаратная поддержка шифрования AES с 128/256-битным ключом протоколы туннелирования: OpenVPN, PPTP, L2TP
DHCP	DHCP-сервер/клиент, DHCP-relay; статические и динамические диапазоны адресов; поддержка Radius; поддержка конфигурируемых опций DHCP.
Утилиты	ping; traceroute; iperf; tcpdump; ssh
Прочие функции	поддержка Radius аутентификации; поддержка Samba; FTP сервер; сервер/клиент синхронизации времени NTP.
Технология PRP	устройство поддерживает протокол бесшовного резервирования PRP IEC 62439-3 в качестве устройства DAN (Double Attached Node), а также может выступать в роли RedBox (Redundancy Box) (только для платформ MX240 и MX681)

Настройка, управление и контроль работы устройства осуществляется с использованием персонального компьютера, подключаемого через сеть Ethernet, либо через консоль (виртуальный COM-порт).

1.5.2 Встроенная система безопасности

Встроенная система информационной безопасности устройства реализует следующие функции:

- идентификация, аутентификация пользователей;

- разделение прав пользователей;
- передача данных с использованием СКЗИ;
- регистрация событий безопасности;
- межсетевое экранирование;
- регистрация событий безопасности и их отправка в централизованные системы мониторинга;
- контроль целостности системы;

Система включает следующие подсистемы:

- подсистема регистрации событий безопасности;
- подсистема проверки целостности;
- подсистема криптозащиты каналов связи;
- подсистема аудита.

Подробное описание работы и конфигурирования системы информационной безопасности устройства приведены в ПЛСТ.421457.100 РП «Контроллер TOPAZ IEC DAS. Руководство пользователя».

1.5.3 Работа кнопок и индикаторов

На передней панели устройства расположены светодиодные индикаторы, отображающие работу устройства. Названия и количество индикаторов зависит от модификации и заказного обозначения устройства.

Также на передней панели устройства расположены кнопки, нажатие на которые осуществляется заостренным предметом.

- Кнопка **RS** предназначена для перезагрузки устройства без отключения питания. Кнопка **RS** может отсутствовать.
- Кнопка **RB** предназначена для активации загрузчика с SD-карты, при одновременном нажатии с кнопкой **RS**. В случае отсутствия кнопки **RS** активация загрузчика с SD-карты осуществляется посредством нажатия кнопки **RB**.

Информация о работе кнопок и индикаторов в различных исполнениях устройства содержится в приложении А.

1.6 Конфигурирование устройства

1.6.1 Подключение к командной строке

Конфигурирование устройства с помощью командной строки возможно через серийную консоль (порт USB на лицевой стороне устройства) либо через порт Ethernet по протоколу ssh.

Таблица 10 – Варианты доступа к настройкам устройства

Протокол	Описание	Требуемое ПО
SSH	Защищенный протокол передачи данных. Аналог протокола Telnet с шифрованием трафика при авторизации и работе с консолью.	UNIX – утилита ssh (стандартный SSH-клиент UNIX); Windows – PuTTY, WinSCP, openssh
Серийная консоль	Подключение через консольный USB-порт устройства (virtual COM-port).	UNIX – утилита minicom; Windows XP – HyperTerminal (встроенное ПО); Windows 7, 8, 10 – PuTTY или аналог

Конфигурирование устройства через SSH-соединение или серийную консоль можно осуществлять с помощью одной из терминальных программ. В приложении Б настоящего РЭ приведен пример подключения к устройству с помощью одной из таких программ.

ВНИМАНИЕ! ПРИ КОНФИГУРИРОВАНИИ УСТРОЙСТВА РЕКОМЕНДУЕТСЯ УДЕЛИТЬ ОСОБОЕ ВНИМАНИЕ НАСТРОЙКАМ ДОСТУПА ПО ПРОТОКОЛУ SSH. ОТ СЛОЖНОСТИ ПАРОЛЕЙ, РАЗРЕШЕНИЯ УДАЛЕННОГО ДОСТУПА, ИСПОЛЬЗУЕМЫХ ПОРТОВ СЕТЕВЫХ СЛУЖБ, НАСТРОЕК МЕЖСЕТЕВОГО ЭКРАНА И ДРУГИХ НАСТРОЕК СЕТЕВЫХ СЛУЖБ ЗАВИСИТ БЕЗОПАСНОСТЬ УСТРОЙСТВА И ПОДКЛЮЧЕННЫХ К НЕМУ УСТРОЙСТВ.

Логин и пароль при заводских настройках следующие:

Логин (Login): **admin**

Пароль (Password): **admin**

Рисунок 1 – Экран приветствия командной строки

1.6.1.1 Подключение через серийную консоль

При подключении устройства через консольный порт (USB) в системе появится виртуальный последовательный COM-порт, который можно использовать для соединения персонального компьютера с устройством. Для того, чтобы узнать номер порта, перейдите в «Диспетчер устройств» Windows и откройте вкладку «Порты». После чего, убедившись, что на устройство подано питание, соедините устройство с компьютером. Во вкладке «Порты» появится новый последовательный порт.

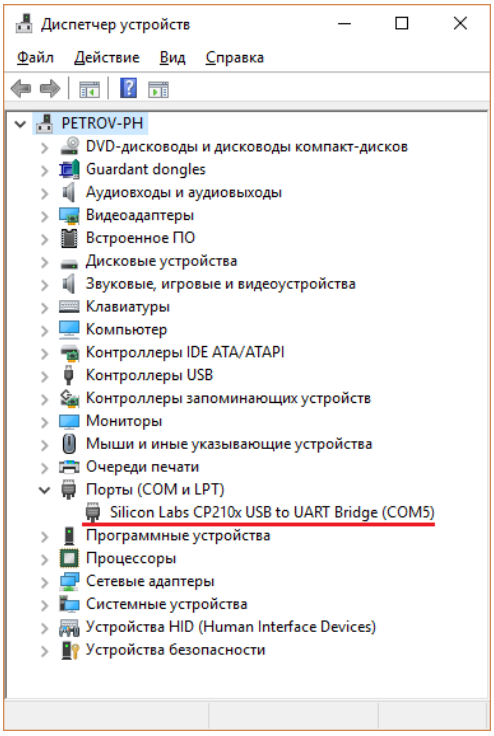


Рисунок 2 – Отображение устройства в диспетчере устройств Windows

Примечание Номер виртуального COM-порта присваивается операционной системой автоматически, поэтому на вашем компьютере он может отличаться от указанного в примере.

Последовательный порт консоли предоставляет пользователю удобный способ подключения к устройству, особенно при первом подключении и настройке устройства. Связь осуществляется по прямому последовательному соединению и пользователю не нужно знать IP адреса Ethernet-портов для того, чтобы подключиться к устройству.

Параметры передачи данных по виртуальному COM-порту приведены в таблице ниже.

Таблица 11 – Параметры соединения с устройством по виртуальному COM-порту

Параметр	Значение
Скорость передачи / Baudrate	115 200 bps
Биты данных / Parity None Data bits	8
Стоповые биты / Stop bits	1
Контроль четности / Parity	None
Управление потоком / Flow Control	None

1.6.1.2 Подключение через порт Ethernet по протоколу SSH

При подключении маршрутизатора к персональному компьютеру через Ethernet используются следующие настройки LAN:

порт LAN#1 192.168.3.127

макса подсети: 255.255.255.0

1.6.2 Команды и утилиты для работы с устройством

Команды консоли, описанные в данном разделе, предназначены для настройки работы, для использования этих команд необходимо войти в режим Shell:

```
shell
```

1.6.2.1 Команда dmesg

Команда **dmesg** предназначена для вывода сообщений ядра системы при загрузке операционной системы.

1.6.2.1.1 Синтаксис

```
dmesg [-c] [-n <уровень>] [-s <размер>]
```

Таблица 12 – Опции команды dmesg

Опция	Описание
-c	Очистить содержимого кольцевого буфера после вывода на экран.
-n <уровень>	Задать *уровень* выводимых сообщений. **-n 1** – выводить только тревожные сообщения
-s <размер>	Использовать буфер заданного *размера* для буфера сообщений. (По умолчанию 16392 байт)

1.6.2.1.2 Пример использования

Вывести на экран последние события ядра и очистить буфер логирования

```
dmesg -c
```

1.6.2.2 Утилита ip

Утилита **ip** предназначена для настройки сетевого интерфейса или для отображения текущей конфигурации.

1.6.2.2.1 Синтаксис

```
ip [ <опции> ] <объект> { <команды> | help }
```

Таблица 13 – Опции утилиты ip

Опция	Описание
-V	Отображение версию утилиты.
-s	Вывести на экран больше информации. Количество повторяющихся опций –s влияет на количество выведенной информации.
-r	Использовать DNS имена вместо адресов хостов.
-f <семейство_прот.>	Задать используемое *семейство протоколов*. На выбор: inet, inet6, bridge, ipx, dnet или link

Таблица 14 – Объекты утилиты ip

Объект	Описание
link	Задать / отобразить сетевой интерфейс
address	Операция с адресом
route	Значение таблицы маршрутизации
rule	Операции с правилами таблицы маршрутизации
neigh	Управление таблицей соседей/ARP

Объект	Описание
tunnel	Настройка туннеля IP
maddress	Добавить / изменить / удалить адрес multicast
mroute	Управление кэшем маршрутизации multicast
monitor	Мониторинг состояния сети
xfrm	Управление политиками IPsec (IP Security)

1.6.2.2.2 Пример использования

Отобразить статус работы всех интерфейсов.

ip link show

Отобразить таблицу правил маршрутизации.

ip route list

Создать правило маршрутизации сетей 192.168.3.0/24 через интерфейс eth0.

ip route add 192.168.3.0/24 dev eth0

Создать правило маршрутизации IP-адреса 192.168.3.1 через шлюз 192.168.1.2.

ip route add 192.168.3.1 via 192.168.1.2

Добавить шлюз по умолчанию 192.168.1.2.

ip route add default via 192.168.1.2

1.6.2.3 Команда logread

Команда **logread** предназначена для вывода сообщений кольцевого буфера syslog.

Синтаксис:

```
logread [-f]
```

Таблица 15 – Опции команды logread

Опция	Описание
-f	Выводить сообщения на экран по мере их появления

1.6.2.3.1 Пример использования

Вывести на экран все сообщения буфа syslog и включить вывод новых сообщений по мере их появления

logread -f

1.6.2.4 Утилита mstpctl

Утилита **mstpctl** предназначена для конфигурирования MST (Multiple Spanning Tree).

1.6.2.4.1 Синтаксис

```
mstpctl [<команда>]
```

Таблица 16 – Команды утилиты mstpctl

Команда	Аргументы	Описание
Команды конфигурирования		
createtree	**<мост> <mstid>**	Создать MSTI (multiple spanning-tree instance) с индексом *msid* для *моста*.

Команда	Аргументы	Описание
deletetree	**<мост> <mstid>**	Удалить MSTI с индексом *msid* для *моста*.
setmaxage	**<мост> <max_age>**	Задать параметр *Max age* для *моста* (20 по умолчанию)
setfdelay	**<мост> <время>**	Задать параметр *время* параметра forward delay для *моста* (15 по умолчанию)
setmaxhops	**<мост> <max_hops>**	Задать параметр *maximum hops* для *моста* (20 по умолчанию)
setforcevers	**<мост> {mstp|rstp|stp}**	Использовать выбранный протокол для *моста* (mstp по умолчанию)
settxholdcount	**<мост> <tx_hold_count>**	Задать параметр *transmit hold count* для *моста*
settreeprio	**<мост> <mstid> <приоритет>**	Задать приоритет моста для дерева с индексом msid. Приоритет – значение между 0 и 15.
setportpathcost	**<мост> <порт> <cost>**	Задать «стоимость» (*cost*) *порта* (0 по умолчанию)
setportadminedge	**<мост> <порт> {yes|no}**	Задать *порт моста* как Edge Port
setportautoedge	**<мост> <порт> {yes|no}**	Включить/отключить автоматическое переключение режима Edge Port для *порта*
setportp2p	**<мост> <порт> {yes|no|auto}**	Включить/отключить режим определения точка-точка (по умолчанию auto)
setportrestrrole	**<мост> <порт> {yes|no}**	Включить/отключить ограничение возможности становиться «корневым» для *порта* (по умолчанию no – без ограничения)
setportrestrtcn	**<мост> <порт> {yes|no}**	Включить/отключить ограничение на распространение полученных оповещений об изменений топологии для *порта* (по умолчанию no – без ограничения)
setbpduguard	**<мост> <порт> {yes|no}**	Включить/отключить функцию **BPDU Guard** (функция, которая позволяет выключать порт при получении BPDU) *порта*. (по умолчанию no – выключена)
settreeportprio	**<мост> <порт> <mstid> <приоритет>**	Задать *приоритет порта* в *мосте* для MSTI с индексом *mstid.* Приоритет – значение между 0 и 15.
sethello	**<мост> <время>**	Задать *время* **Hello BPDU** *порта*. (2 по умолчанию)
setageing	**<мост> <время>**	(только STP) Задать время aging-time в секундах (300 по умолчанию)

Команда	Аргументы	Описание
setportnetwork	**<мост> <порт> {yes|no}**	Включить/отключить функцию **Bridge Assurance** для данного *порта*
Команды отображения		
showbridge	**[<мост>]**	Отобразить информацию о топологии CIST *моста*
showport	**<мост> [<порт>]**	Отобразить краткую информацию о топологии CIST *порта* данного *моста*
showportdetail	**<мост> [<порт>]**	Отобразить детальную информацию о топологии CIST *порта* данного *моста*
showtree	**<мост> <mstid>**	Отобразить информацию о MST с индексом *mstid* для *моста*
showtreeport	**<мост> <порт> <mstid>**	Отобразить детальную информацию о MST с индексом *mstid* для *порта* данного *моста*

1.6.2.5 Утилита netstat

Утилита **netstat** предназначена для отображения информации о сети.

1.6.2.5.1 Синтаксис

```
netstat [<опции>]
```

Таблица 17 – Опции команды netstat

Опция	Описание
-l [<интерфейс>]	Отобразить сокеты прослушивателя. Сокет - программный интерфейс для обеспечения обмена данными между процессами.
-a	Отобразить все сокеты
-e	Отобразить больше информации
-n	Показывать сетевые адреса как числа.
-r	Отобразить таблицы маршрутизации
-t	Отобразить сокеты TCP
-u	Отобразить сокеты UDP
-w	Отобразить сокеты RAW
-x	Отобразить сокеты UNIX

1.6.2.5.2 Пример использования

Отобразить сокеты TCP.

```
netstat -t
```

1.6.2.6 Команда passwd

Команда **passwd** предназначена для изменения пароля учетной записи.
Пароль может состоять из букв английского алфавита и цифр.

После ввода команды и нажатия клавиши Enter необходимо дважды ввести новый пароль. По завершению в консоли отобразится сообщение о том, что пароль был изменен, как показано на рисунке ниже.

```
root@TOPAZ:~# passwd
Changing password for root
New password:
Retype password:
passwd: password for root changed by root
```

Рисунок 3

Примечание При заводских настройках во время авторизации так же появится предупреждение об уязвимости системы по причине отсутствия пароля авторизации, как показано на рисунке 8.

```
=== WARNING! ======================================
There is no root password defined on this device!
Use the "passwd" command to set up a new password
in order to prevent unauthorized SSH logins.
--------------------------------------------------
```

Рисунок 4

1.6.2.7 Команда ping и ping6

Команда **ping (ping6)** предназначена для отправки ICMP эхо-запроса на указанный хост.

1.6.2.7.1 Синтаксис

```
ping [-c <NN>] [-s <размер>] [-q] <хост> [-I <интерфейс>] <интерфейс>
ping6 [-c <NN>] [-s <размер>] [-q] <хост> [-I <интерфейс>] <интерфейс>
```

Таблица 18 – Опции команды ping (ping6)

Опция	Описание
-c <NN>	Послать *NN* запросов
-s <размер>	Послать объем данных указанного *размера* (по умолчанию 56 байт)
-q	«Тихий режим», выводит на экран информацию во время начала посылки данных и по завершению.
-I <интерфейс>	Выбрать исходящий *интерфейс*

1.6.2.7.2 Пример использования

Отправить IPv4 эхо-запрос в виде одного ICMP пакета размером 500 В на адрес 10.0.0.1.

```
ping -c 1 -s 500 10.0.0.1
```

1.6.2.8 Утилита service

Утилита `service` предназначена для запуска, перезагрузки и остановки сервисов. Что бы узнать имя сервиса, введите данную команду без аргументов. На экране будет отображен список всех сервисов.

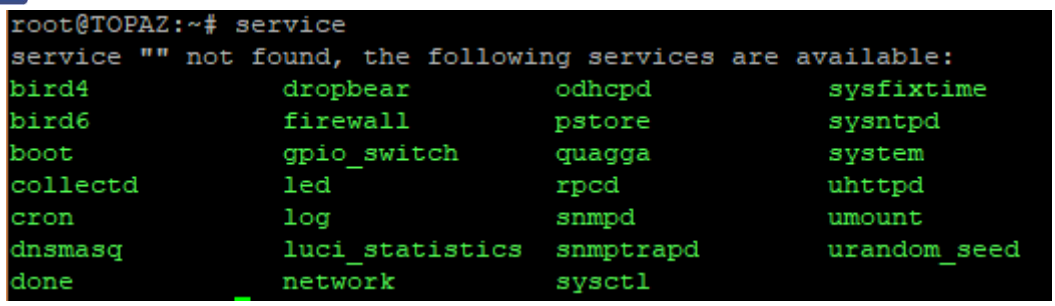


Рисунок 5 – Список запущенных сервисов

1.6.2.8.1 Синтаксис

```
service [<сервис> <команда>]
```

Таблица 19 – Опции команды service

Команды	Описание
start	Запуск сервиса
stop	Остановка сервиса
restart	Перезапуск сервиса
reload	Обновление конфигурации сервиса (Для применения изменений конфигурации устройства без перерыва в работе)
enable	Разрешить сервис
disable	Запретить сервис

1.6.2.8.2 Пример использования

Обновление конфигурации сервиса firewall.

service firewall reload

1.7 VRF

Для создания любых интерфейсов необходимо войти в меню System:

system

Потом запустить режим конфигурации:

configure terminal

Далее для создания VRF:

vrf interface table-id <VRF table id>

<VRF table id> - уникальный идентификатор таблицы VRF, число от 1 до 252 (идентификаторы от 253 до 255 зарезервированы). Б создан интерфейс **vrf<VRF table id>**, и открыто подменю интерфейса VRF со строкой приглашения вида **config-vrf<VRF table id>.**

В подменю интерфейса VRF можно задать интерфейсы для указанного VRF:

interface <Network interface>

<Network interface> - имя сетевого интерфейса, по нажатию “TAB” будут предложены возможные варианты.

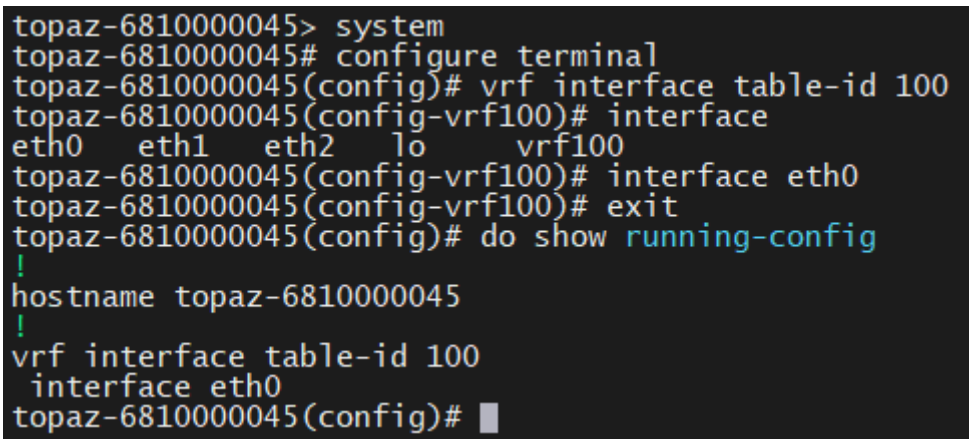


Рисунок 6

Для удаления интерфейса из VRF, в подменю **config-vrf<VRF table id>** ввести команду **no interface <Network interface>**

Для удаления VRF ввести команду **no vrf interface table-id <VRF table id>**

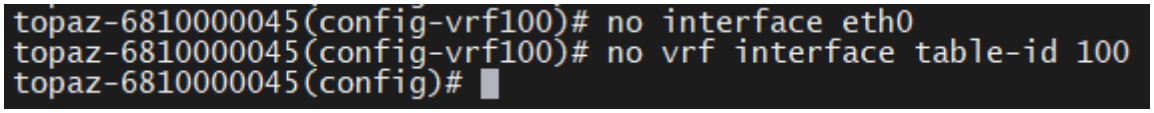


Рисунок 7

Создание интерфейсов

1.7.1 GRE

Для создания GRE туннеля запустить режим конфигурации:
configure terminal

Дать команду
tunnel tun-id <Tunnel id> mode <GRE mode>

<Tunnel id> - номер-идентификатор интерфейса от 1 до 255, <GRE mode> - режим работы туннеля, **gre** или **gretap**, в режиме gretap туннель может работать с кадрами канального уровня, интерфейс можно добавить в bridge.

В результате исполнения команды будет создан туннельный интерфейс **tun<Tunnel id>**, и открыто подменю интерфейса со строкой приглашения вида **config-if-tun<Tunnel id>.**

В подменю интерфейса необходимо задать локальный IP-адрес и IP-адрес удалённой стороны:
local <A.B.C.D>
remote <A.B.C.D>

Здесь же можно настроить ключ:
key <Key>

<Key> - 32-битное число, одинаковое для всех участников тоннеля.
Можно задать установку TTL для туннелированных пакетов:
ttl <TTL value>

Команда **no ttl** отключит эту опцию, TTL туннелированных пакетов не будет изменяться.

```
topaz-6810000045# configure terminal
topaz-6810000045(config)# tunnel tun-id 1 mode gre
topaz-6810000045(config-if-tun1)#local 10.10.10.10
topaz-6810000045(config-if-tun1)# remote 20.20.20.20
topaz-6810000045(config-if-tun1)# ttl 64
topaz-6810000045(config-if-tun1)# key 0xffffffff
topaz-6810000045(config-if-tun1)# exit
topaz-6810000045(config)# do show running-config
!
hostname topaz-6810000045
!
tunnel tun-id 1 mode gre
 key 0xffffffff
 local 10.10.10.10
 remote 20.20.20.20
 ttl 64
!
```

Рисунок 8

Для удаления туннельного интерфейса используется команда **no tunnel tun-id <Tunnel id>**

```
topaz-6810000045(config-if-tun1)# no tunnel tun-id 1
topaz-6810000045(config)#
```

Рисунок 9

Настройка IP-адресации производится в меню **router**.

1.7.2 Loopback

Запустить режим конфигурации:
configure terminal

Дать команду:
loopback id <lo id>

<lo id> - номер-идентификатор loopback-интерфейса.

```
topaz-6810000045(config)# loopback id 10
topaz-6810000045(config-lo10)# do show running-config
!
hostname topaz-6810000045
!
loopback id 10
```

Рисунок 10

Для удаления loopback-интерфейса используется команда **no loopback id <lo id>**

```
topaz-6810000045(config-lo10)# no loopback id 10
topaz-6810000045(config)#
```

Рисунок 11

Настройка IP-адресации производится в меню **router**.

1.7.3 VLAN

Запустить режим конфигурации:

configure terminal

Дать команду:

vlan interface <ethernet/bridge> <Network interface> vlan-id <VLAN id>

<ethernet/bridge> - выбор типа интерфейса, <Network interface> - имя сетевого интерфейса, к которому будет привязан VLAN-интерфейс, по нажатию “TAB” будут показаны доступные варианты, <VLAN id> - идентификатор VLAN. В результате исполнения команды будет создан VLAN-интерфейс **<Network interface>.<VLAN id>**.

```
topaz-6810000045# configure terminal
topaz-6810000045(config)# vlan interface
ethernet bridge
topaz-6810000045(config)# vlan interface ethernet
eth1 eth2 eth0
topaz-6810000045(config)# vlan interface ethernet eth0 vlan-id
  Number in the range 1-4094  VLAN id

topaz-6810000045(config)# vlan interface ethernet eth0 vlan-id 10
topaz-6810000045(config-if-eth0.10)# do show running-config
!
hostname topaz-6810000045
!
vlan interface ethernet eth0 vlan-id 10
topaz-6810000045(config-if-eth0.10)#
```

Рисунок 12

Для удаления VLAN-интерфейса используется команда **no vlan interface <ethernet/bridge> <Network interface> vlan-id <VLAN id>**

```
topaz-6810000045(config-if-eth0.10)# no vlan interface ethernet eth0 vlan-id 10
topaz-6810000045(config)#
```

Рисунок 13

Настройка IP-адресации производится в меню **router**.

1.7.4 Bridge

Запустить режим конфигурации:

configure terminal

Дать команду:

bridge interface group-id <Bridge group id>

<Bridge group id> - номер-идентификатор интерфейса от 1 до 255. В результате работы команды будет создан bridge-интерфейс **br<Bridge group id>**, и открыто подменю интерфейса со строкой приглашения вида **config-if-br<Bridge group id>.** В подменю интерфейса можно добавить интерфейсы в этот bridge:

interface <ethernet/tunnel> <Network interface>

<ethernet/tunnel> - выбор типа интерфейса, <Network interface> - имя сетевого интерфейса, по нажатию “TAB” будут показаны доступные варианты.

Можно включить Spanning Tree Protocol:

stp <classic/rapid>

Отключить Spanning Tree Protocol можно командой **no stp**

```
topaz-6810000045(config)# bridge interface group-id 1
topaz-6810000045(config-if-br1)# interface
ethernet tunnel
topaz-6810000045(config-if-br1)# interface tunnel tun1
topaz-6810000045(config-if-br1)# interface ethernet
eth1 eth2 eth0
topaz-6810000045(config-if-br1)# interface ethernet eth1
topaz-6810000045(config-if-br1)# stp
classic rapid
topaz-6810000045(config-if-br1)# stp rapid
topaz-6810000045(config-if-br1)# do show running-config
!
hostname topaz-6810000045
!
tunnel tun-id 1 mode gretap
 local 10.10.10.10
 remote 20.20.20.20
 ttl 64
!
bridge interface group-id 1
 interface ethernet eth1
 interface tunnel tun1
 stp rapid
topaz-6810000045(config-if-br1)#
```

Рисунок 14

Удалить bridge-интерфейс можно командой **no bridge interface group-id <Bridge group id>**

```
topaz-6810000045(config-if-br1)# no bridge interface group-id 1
topaz-6810000045(config)#
```

Рисунок 15

Настройка IP-адресации производится в меню **router**.

1.8 Подсистема router

1.8.1 FRR

FRR предоставляет услуги IP-маршрутизации. Его роль в сетевом стеке заключается в обмене информацией о маршрутизации с другими маршрутизаторами, принятии решений о маршрутизации и политике, а также информировании других уровней об этих решениях. В наиболее распространенном сценарии FRR устанавливает решения о маршрутизации в ядро операционной системы, позволяя сетевому стеку ядра принимать соответствующие решения о пересылке.

В дополнение к динамической маршрутизации FRR поддерживает полный спектр конфигурации L3, включая статические маршруты, адреса, рекламные объявления маршрутизатора и т. д. Он также имеет некоторые легкие функции L2, но в основном это остается

за подсистемой Linux. Это делает его подходящим для развертываний, начиная от небольших домашних сетей со статическими маршрутами и заканчивая интернет-обменниками с полными интернет-таблицами.

1.8.2 Basic Commands

1.8.2.1 Команды настройки

В конфигурационном файле вы можете записать параметры отладки, пароль vty, конфигурации демона маршрутизации, имя файла журнала и так далее. Эта информация формирует начальный набор команд для зверя маршрутизации при его запуске.

Файлы конфигурации находятся в /etc/frr.

1.8.2.2 Методы настройки

Существует два способа настройки FRR.

Обычно у каждого из демонов есть свой собственный конфигурационный файл. По умолчанию имя конфигурационного файла представлено в следующем виде имя демона.conf. Например, конфигурационный файл zebra по умолчанию был zebra.conf. Этот метод устарел.

Из-за большого количества файлов конфигурации, которые это создает, и склонности одного демона полагаться на другие для определенных функций, большинство развертываний теперь используют “интегрированную” конфигурацию. При такой настройке вся конфигурация, как правило, помещается в один файл `/etc/frr/frr.conf`.

При запуске FRR с помощью сценария инициализации или systemd `vtysh` вызывается для чтения файла конфигурации и отправки соответствующих фрагментов только заинтересованным в них демонам. Запущенные обновления конфигурации сохраняются обратно в этот единственный файл с помощью `vtysh`. Это рекомендуемый метод. Чтобы использовать этот метод, добавьте следующую строку в `/etc/frr/vtysh.conf`:

service integrated-vtysh-config

Если вы установили из исходного кода или использовали пакет, то, вероятно, данная строка уже присутствует.

При желании вы можете указать файл конфигурации, используя параметры `-f` или `--config_file` при запуске демона.

1.8.2.3 Основные команды конфигурации

hostname HOSTNAME

Задайте имя хоста маршрутизатора. Это только для текущего `vtysh`, оно не будет сохранено ни в один файл конфигурации даже с `write file`.

domainname DOMAINNAME

Задайте доменное имя маршрутизатора. Это только для текущего `vtysh`, оно не будет сохранено ни в один файл конфигурации даже с `write file`.

password PASSWORD

Установите пароль для интерфейса vty. `no` Форма команды удаляет пароль. Если пароль отсутствует, vty не будет принимать подключения.

enable password PASSWORD

Установите пароль для включения. `no` форма команды удаляет пароль для включения.

service cputime-stats

Сбор статистики использования ЦП для отдельных обработчиков событий FRR и команд командной строки. Это включено по умолчанию и может быть отключено, если дополнительные накладные расходы вызывают заметное замедление работы вашей системы.

Отключение этой статистики также сделает `service cputime-warning (1-4294967295)` ограничение неработоспособным.

service cputime-warning (1-4294967295)

Предупреждать, если загрузка процессора обработчиком событий или командой командной строки превышает указанный предел (в миллисекундах). Такие предупреждения обычно указывают на то, что какая-то процедура в FRR ошибочно блокирует / блокирует цикл обработки, и о ней следует сообщать как об ошибке FRR.

Ограничение по умолчанию составляет 5 секунд (т.е. 5000), но это может быть изменено с помощью устаревшего `--enable-time-check=...` параметра времени компиляции.

Эта команда не имеет никакого эффекта, если `service cputime-stats` она отключена.

service walltime-warning (1-4294967295)

Предупреждать, если общее время работы wallclock, затраченное на обработку события или выполнение команды CLI, превышает указанный предел (в миллисекундах).) Сюда входит время, затраченное на ожидание ввода-вывода или выполнения других задач, и может выдавать чрезмерные предупреждения, если система перегружена. (Это все еще может быть полезно для немедленного указания на то, что FRR работает некорректно из-за вызванного извне голодания.)

Ограничение по умолчанию составляет 5 секунд, как указано выше, включая тот же устаревший `--enable-time-check=...` параметр времени компиляции.

log trap LEVEL

Эти команды устарели и присутствуют только для исторической совместимости. Команда log trap устанавливает текущий уровень ведения журнала для всех включенных назначений ведения журнала и устанавливает значение по умолчанию для всех будущих команд ведения журнала, которые не указывают уровень. Обычным уровнем ведения журнала по умолчанию является отладка. `no` форма команды сбрасывает уровень по умолчанию для будущих команд ведения журнала на отладочный, но не изменяет уровень ведения журнала для существующих назначений ведения журнала.

log stdout LEVEL

Включите вывод журнала в стандартный вывод. Если необязательный второй аргумент, указывающий уровень ведения журнала, отсутствует, будет использоваться уровень ведения журнала по умолчанию (обычно отладка). `no` Форма команды отключает ведение журнала в стандартный вывод. `LEVEL` Аргумент должен иметь одно из следующих значений: аварийные ситуации, предупреждения, критические, ошибки, предупреждения, уведомления, информационные или отладочные. Обратите внимание, что существующий код регистрирует свои наиболее важные сообщения со строгостью `errors`.

Примечание Если systemd используется и стандартный вывод подключен к systemd, FRR автоматически переключится на journaldрасширенное ведение журнала для этой цели.

Примечание FRRouting использует writev() системный вызов для записи сообщений журнала.
Предполагается, что этот вызов является атомарным, но на самом деле это не относится к каналам или терминалам, а только к обычным файлам. Это означает, что в редких случаях одновременные сообщения журнала из разных потоков могут перемешиваться в выводе терминала. Используйте

файл журнала и tail -f, если эта редкая возможность неприемлема для вашей установки.

log file [FILENAME [LEVEL]]

Если вы хотите войти в файл, пожалуйста, укажите `filename`, как в этом примере:

log file /var/log/frr/bgpd.log informational

Если необязательный второй аргумент, указывающий уровень ведения журнала, отсутствует, будет использоваться уровень ведения журнала по умолчанию (обычно отладочный, но может быть изменен с помощью, устаревшей `log trap` команды). `no` форма команды отключает запись в файл.

log syslog [LEVEL]

Включить вывод журнала в системный журнал. Если необязательный второй аргумент, указывающий уровень ведения журнала, отсутствует, будет использоваться уровень ведения журнала по умолчанию (обычно отладочный, но может быть изменен с помощью, устаревшей log traпкоманды). noФорма команды отключает ведение журнала в системный журнал.

Примечание При этом используется системный syslog()API, который не поддерживает пакетную обработку сообщений или структурированные пары данных ключ / значение. Если возможно, используйте log extended EXTLOGNAMEwith destination syslog [supports-rfc5424] вместо this..

log extended EXTLOGNAME

Создайте расширенную цель ведения журнала с указанным именем. Имя не имеет дальнейшего значения и используется только для идентификации цели. С помощью формы можно создавать и удалять несколько целевых no объектов.

Обратитесь к расширенной цели ведения журнала для получения дополнительной информации и подразделов.

log monitor[LEVEL]

Эта команда устарела и ничего не делает.

log facility[FACILITY]

Эта команда изменяет средство, используемое в сообщениях системного журнала. По умолчанию используется `daemon`. `no` Форма команды сбрасывает средство на средство по умолчанию `daemon`.

log record-priority

Чтобы включить серьезность во все сообщения, записанные в файл, в стандартный вывод или в монитор терминала (т. Е. Во все, кроме системного журнала), используйте команду `log record-priority` глобальной конфигурации. Чтобы отключить эту опцию, используйте `no` форму команды. По умолчанию уровень серьезности не включается в зарегистрированные сообщения. Примечание: некоторые версии syslogd можно настроить для включения объекта и уровня в отправляемые сообщения.

log timestamp precision[(0-6)]

Эта команда устанавливает точность временных меток сообщений журнала с заданным количеством цифр после запятой. В настоящее время значение должно находиться в диапазоне от 0 до 6 (т. Е. Максимальная точность составляет микросекунды). Чтобы восстановить поведение по умолчанию (точность в 1 секунду), используйте noформу команды или явно задайте точность в 0.

log timestamp precision 3

В этом примере точность установлена для предоставления временных меток с точностью до миллисекунд.

log commands

Эта команда позволяет записывать все команды, введенные пользователем, во все разрешенные пункты назначения журнала. Обратите внимание, что ведение журнала включает полные командные строки, включая пароли. Если для запуска демона используется параметр запуска демона *–command-log-always*, то эта команда включена по умолчанию и не может быть отключена, а форма [нет] команды запрещена.

log filtered-file [FILENAME [LEVEL]]

Настройте целевой файл для отфильтрованных журналов с `log filter-text WORD` помощью команды.

log filter-text WORD ¶

Эта команда принудительно фильтрует журналы по определенной строке. Сообщение журнала будет напечатано, только если оно соответствует одному из фильтров в таблице фильтров журнала. Фильтр применяется только к целевым объектам ведения журнала файлов, настроенным с `log filtered-file [FILENAME [LEVEL]]` помощью .

Примечание Фильтры журналов помогают, когда вам нужно включить отладку, которая вызывает значительную нагрузку на систему (включение определенных отладок может привести к остановке FRR). Фильтры журналов предотвращают это, но вы все равно должны ожидать небольшого снижения производительности из-за фильтрации каждого из всех этих журналов.

Внимание Этот параметр не сохраняется frr.confи не отображается show running-config. Он предназначен только для кратковременных целей отладки.

clear log filter-text

Эта команда очищает все текущие фильтры в таблице фильтров журнала.

log immediate-mode

Используйте небуферизованный вывод для сообщений журнала и отладки; обычно существует некоторая внутренняя буферизация.

log unique-id

Включите [XXXXX-XXXXX] уникальный идентификатор сообщения журнала в текстовую часть сообщений журнала. Это включено по умолчанию, но может быть отключено с no log unique-idпомощью. Пожалуйста, убедитесь, что идентификаторы включены при включении журналов для отчетов об ошибках FRR.

Уникальные идентификаторы генерируются автоматически на основе имени файла исходного кода, строки формата (перед заполнением) и серьезности. Они не меняются “случайным образом”, но некоторые работы по очистке могут привести к большим изменениям идентификаторов между выпусками. Идентификаторы всегда начинаются с буквы, состоят из букв и цифр (и тире для удобства чтения), не чувствительны к регистру и I, L, O& Uисключены.

Этот параметр не повлияет на будущие цели ведения журнала, которые позволяют помещать уникальный идентификатор во вспомогательные метаданные вне текстового содержимого сообщения журнала. (В настоящее время такой цели ведения журнала не существует, но RFC5424 syslog и журнал systemd поддерживают его.)

debug unique-id XXXXX-XXXXX backtrace

Распечатайте обратные трассировки (стек вызовов) для определенных сообщений журнала, идентифицируемых по их уникальному идентификатору (см. Выше).) Включает местоположение исходного кода и выполняемый текущий обработчик событий. В некоторых системах может потребоваться установить пакет символов отладки, чтобы получить правильные имена функций, а не указатели на исходный код.

Эта команда может быть выдана как в режиме конфигурации, так и вне его, и сохраняется в конфигурации, только если она была задана в режиме конфигурации.

Примечание Печать обратных трассировок может значительно замедлить ведение журнала вызовов и привести к быстрому увеличению размера файлов журнала. Не забудьте отключить обратные трассировки, когда они больше не нужны.

service password-encryption

Зашифруйте пароль.

service advanced-vty

Включите VTY в расширенном режиме.

service terminal-length (0-512)

Установите общесистемную конфигурацию линии. Эта команда настройки применяется ко всем интерфейсам VTY.

line vty

Войдите в режим настройки vty.

banner motd default

Установите строку motd по умолчанию.

banner motd file FILE

Установите строку motd из файла. Файл должен находиться в каталоге, указанном ниже `--sysconfdir`.

banner motd

Задайте строку motd из входных данных.

exec-timeout MINUTE [SECOND]

Установите значение тайм-аута подключения VTY. Когда указан только один аргумент, он используется для значения времени ожидания в минутах. Необязательный второй аргумент используется для значения времени ожидания в секундах. Значение тайм-аута по умолчанию составляет 10 минут. Когда значение тайм-аута равно нулю, это означает отсутствие тайм-аута.

Если не установить это значение или не установить значения 0 0, тайм-аут не будет включен.

access-class ACCESS-LIST

Ограничение подключений vty с помощью списка доступа.

allow-reserved-ranges

Разрешить использование IP-адресов, зарезервированных для IP-адресов IPv4 (класс E) для демонов. Например, установка IPv4-адресов для интерфейсов или разрешение зарезервированных диапазонов в следующих переходах BGP.

По умолчанию: выключено.

1.8.2.4 Пример конфигурационного файла

Ниже приведен пример файла конфигурации для демона zebra.

```
!
```

```
!
!
!
frr version 6.0
!
frr defaults traditional
!
log stdout
!
!
```

! и # являются символами комментариев. Если первый символ слова является одним из символов комментария, то дальнейшая строка будет игнорироваться как комментарий.

1.8.2.5 Управление версиями конфигурации, профили и поведение при обновлении

Все демоны frr совместно используют механизм указания профиля конфигурации и версии для загрузки и сохранения конфигурации. Конкретные параметры конфигурации принимают разные значения по умолчанию в зависимости от выбранного профиля и версии.

В то время как профиль может быть выбран пользовательской конфигурацией и останется при обновлении, frr всегда будет записывать конфигурации, используя свою текущую версию. Это означает, что после обновления `write file` может выписать конфигурацию, немного отличающуюся от той, что была прочитана.

Поскольку предыдущая конфигурация загружается со значениями по умолчанию для своей версии, но новая конфигурация записывается с новыми значениями по умолчанию, любое значение по умолчанию, которое изменилось между версиями, приведет к записи соответствующей записи конфигурации. Настройка маршрутизации является сложной и остается неизменной при обновлении. Измененные значения по умолчанию повлияют только на новую конфигурацию.

Обратите внимание, что загруженная версия сохраняется в сеансах интерактивной настройки. Команды, выполняемые в сеансе интерактивной настройки, ничем не отличаются от конфигурации, загруженной при запуске. Это означает, что, когда, скажем, вы настраиваете новый одноранговый узел BGP, для настройки используются значения по умолчанию, выбранные последней `frr version` командой.

Примечание Сохранение конфигурации не перенаправляет демонов на использование новой версии по умолчанию, но их перезапуск приведет к этому, поскольку затем они применят новую `frr version` команду, которая была записана. Вручную выполните `frr version` команду в `show running-config`, чтобы избежать этого промежуточного состояния.

Это видно в `show running-config`:

```
Current configuration:
!
! loaded from 6.0
frr version 6.1.1-dev
frr defaults traditional

!
```

Если вы сохраните и затем перезапустите эту конфигурацию, старые значения по умолчанию больше не будут применяться. Аналогичным образом, вы могли бы выполнить `frr version 6.1-dev`, в результате чего применяются новые значения по умолчанию и `loaded from 6.0` комментарий исчезнет.

1.8.2.6 **Профили**

frr предоставляет профили конфигурации для адаптации настроек по умолчанию к различным сценариям использования. В настоящее время реализованы следующие профили:

`traditional` - отражает значения по умолчанию, соответствующие в основном стандартам IETF или общепринятым практикам глобальной интернет-маршрутизации.

`datacenter` - отражает единый административный домен с внутридоменными ссылками, использующими агрессивные таймеры.

Ваш дистрибутив / установка может предварительно установить профиль с помощью опции `-F` командной строки для всех демонов. Все демоны должны быть настроены для одного и того же профиля. Значение, указанное в командной строке, является только предварительно установленным, и любой `frr defaults` оператор в конфигурации будет иметь приоритет.

Примечание Профиль должен быть одинаковым для всех демонов. Несоответствия могут привести к неопределенному поведению.

Вы можете свободно переключаться между профилями, не вызывая никаких прерываний или изменений конфигурации. Все настройки остаются на своих предыдущих значениях, и `show running-configuration` на выходе будет новый вывод, содержащий предыдущие значения по умолчанию в виде явной конфигурации. Новая конфигурация, например, добавление узла BGP, будет использовать новые значения по умолчанию. Чтобы применить новые значения по умолчанию для существующей конфигурации, ранее невидимые старые значения по умолчанию, которые теперь отображаются, должны быть удалены из конфигурации.

1.8.2.7 **Методы обновления для автоматически созданной конфигурации**

При использовании frr с сгенерированными конфигурациями (например, Ansible, Puppet и т. Д.) Соображения по обновлению несколько отличаются:

1. Всегда записывайте `frr version` инструкцию в создаваемых конфигурациях. Это гарантирует, что значения по умолчанию применяются последовательно.
2. Старайтесь не запускать больше разных версий frr, чем необходимо. Возможно, потребуется проверить каждую версию по отдельности. Если выполняется сочетание старых и новых установок, используйте самую старую версию для `frr version` инструкции.
3. При развертывании обновлений создайте конфигурацию, как обычно, с идентификатором старой версии и загрузите ее. Проверьте, нет ли каких-либо различий или предупреждений об устаревании. Если в конфигурации есть различия, передайте их обратно в генератор конфигурации, чтобы свести к минимуму зависимость от фактических значений по умолчанию.
4. После удаления последней установки старой версии измените генерацию конфигурации на более новую `frr version`, если это необходимо. Выполните те же проверки, что и при развертывании обновлений.

1.8.2.8 Команды терминального режима

write terminal

Отображает текущую конфигурацию в интерфейсе vty.

write file

Запишите текущую конфигурацию в файл конфигурации.

configure [terminal]

Перейдите в режим конфигурации. Эта команда является первым шагом к настройке.

terminal length(0-512)

Установите длину отображения терминала на (0-512). Если длина равна 0, управление отображением не выполняется.

who

Отображение списка подключенных в данный момент сеансов vty.

list

Перечислите все доступные команды.

show version

Отображение текущей версии frr и информации о хосте сборки.

show logging

Показывает текущую конфигурацию системы ведения журнала. Это включает в себя статус всех назначений ведения журнала.

show log-filter

Показывает текущие фильтры журнала, примененные к каждому демону.

show memory [DAEMON]

Показать информацию о том, сколько памяти используется для каких конкретных вещей в frr. Выходные данные могут отличаться в зависимости от возможностей системы, но в целом они будут выглядеть как показано на рисунке ниже:

```
frr# show memory
System allocator statistics:
  Total heap allocated:  1584 KiB
  Holding block headers: 0 bytes
  Used small blocks:     0 bytes
  Used ordinary blocks:  1484 KiB
  Free small blocks:     2096 bytes
  Free ordinary blocks:  100 KiB
  Ordinary blocks:       2
  Small blocks:          60
  Holding blocks:        0
(see system documentation for 'mallinfo' for meaning)
--- qmem libfrr ---
Buffer                         :          3      24                 72
Buffer data                    :          1    4120               4120
Host config                    :          3  (variably sized)       72
Command Tokens                 :       3427      72             247160
Command Token Text             :       2555  (variably sized)    83720
Command Token Help             :       2555  (variably sized)    61720
Command Argument               :          2  (variably sized)       48
Command Argument Name          :        641  (variably sized)    15672
[...]
--- qmem Label Manager ---
--- qmem zebra ---
ZEBRA VRF                      :          1     912                920
Route Entry                    :         11      80                968
Static route                   :          1     192                200
RIB destination                :          8      48                448
RIB table info                 :          4      16                 96
Nexthop tracking object        :          1     200                200
Zebra Name Space               :          1     312                312
--- qmem Table Manager ---
```

Рисунок 16

Чтобы получить представление о статистике системного распределителя, обратитесь к справочной странице вашей системы mallinfo(3).

Под этой статистикой печатается статистика по отдельным типам выделения памяти в frr (так называемые MTYPEs):

- первый столбец чисел - это текущее количество выделений, сделанных для типа (число уменьшается при освобождении элементов).
- второй столбец - это размер каждого элемента. Это доступно только в том случае, если выделения для типа всегда выполняются с одинаковым размером.
- третий столбец - это общий объем памяти, выделенный для определенного типа, включая заполнение, применяемое malloc. Это означает, что число может быть больше, чем первый столбец, умноженный на второй. Накладные расходы, связанные с бухгалтерией malloc, не включены в это, и столбец может отсутствовать, если системная поддержка недоступна.

При выполнении этой команды из `vtysh`, использование памяти каждого из демонов печатается последовательно. Вы можете указать имя демона, чтобы выводить только его использование памяти.

show motd

Показать текущий баннер motd.

show history

Сбросьте историю vtysh cli.

logmsg

Отправьте сообщение всем адресатам ведения журнала, которые включены для сообщений заданной серьезности.

ind REGEX...

Эта команда выполняет поиск по регулярным выражениям по всем определенным командам во всех режимах. В качестве примера предположим, что вы находитесь в режиме включения и не можете вспомнить, где находится команда для включения маршрутизации сегментов OSPF:

```
frr# find segment-routing on
  (ospf)  segment-routing on
  (isis)  segment-routing on
```

Режим командной строки отображается рядом с каждой командой. В этом примере `segment-routing on` находится в режиме ospf маршрутизатора.

Аналогично, предположим, вам нужен список всех команд, содержащих “l2vpn” и “neighbor”:

```
frr# find l2vpn.*neighbor
  (view)  show [ip] bgp l2vpn evpn neighbors <A.B.C.D|X:X::X:X|WORD> advertised-
routes [json]
  (view)  show [ip] bgp l2vpn evpn neighbors <A.B.C.D|X:X::X:X|WORD> routes [json]
  (view)  show [ip] bgp l2vpn evpn rd ASN:NN_OR_IP-ADDRESS:NN neighbors
<A.B.C.D|X:X::X:X|WORD> advertised-routes [json]
  (view)  show [ip] bgp l2vpn evpn rd ASN:NN_OR_IP-ADDRESS:NN neighbors
<A.B.C.D|X:X::X:X|WORD> routes [json]
  ...
```

Обратите внимание, что при вводе пробелов как части спецификации регулярных выражений повторяющиеся пробелы будут сжаты в один пробел для целей сопоставления. Это является следствием использования пробелов для разграничения токенов CLI. Если вам нужно сопоставить более одного пробела, используйте `\s` escape.

Поддерживаются расширенные регулярные выражения POSIX.

show thread cpu [r|w|t|e|x]

Эта команда отображает статистику выполнения системы для всех различных типов событий. Если параметр не указан, все различные типы запуска отображаются вместе. Кроме того, вы можете попросить посмотреть (r) ead, (w) rite, (t) imer, (e) vent и e (x) типы событий потока ecute. Если вы скомпилировали с помощью disable-cpu-time, эта команда не будет отображаться.

show thread poll

Эта команда отображает данные опроса FRR. Это позволяет взглянуть на то, как мы устанавливаем каждый отдельный fd для команды опроса в данный момент времени.

show thread timers

Эта команда отображает данные таймера FRR для таймеров, которые будут появляться в будущем.

show yang operational-data XPATH [{format <json|xml>|translate TRANSLATOR|with-config}] DAEMON

Отображение операционных данных YANG, начиная с XPATH. Формат по умолчанию - JSON, но также может отображаться в формате XML.

Обычно рабочие данные YANG находятся внутри контейнеров, помеченных как доступные только для чтения.

При желании также можно отобразить листы конфигурации в дополнение к рабочим данным с помощью опции with-config. Этот параметр позволяет отображать листы конфигурации с их текущим настроенным значением (если лист является необязательным, он будет отображаться только в том случае, если он был создан или имеет значение по умолчанию).

1.8.2.9 Общие параметры вызова

Эти параметры применяются ко всем демонам frr.

-d, --daemon

Запуск в режиме демона.

-f, --config_file <файл>

Задайте имя файла конфигурации.

-h, --help

Отобразите эту справку и завершите работу.

-i, --pid_file <файл>

При запуске идентификатор процесса демона записывается в файл, обычно в `/var/run`.

Этот файл может использоваться системой инициализации для реализации таких команд, как `.../init.d/zebra status`, `.../init.d/zebra restart` или `.../init.d/zebra stop`.

Имя файла - это параметр времени выполнения, а не параметр времени настройки, чтобы одновременно можно было запускать несколько демонов маршрутизации. Это полезно при использовании frr для реализации looking glass маршрутизации. Одна машина может использоваться для сбора различных представлений маршрутизации из разных точек сети.

-A , --vty_addr <address>

Задайте локальный адрес VTY для привязки. Если задано, сокет VTY будет привязан только к этому адресу.

-P , --vty_port <port>

Задайте номер TCP-порта VTY. Если установлено значение 0, то сокеты TCP VTY не будут открыты.

-u <user>

Установите пользователя и группу для запуска как.

-N <namespace>

Задайте пространство имен, в котором будет выполняться демон. “/<пространство имен>” будет добавлено ко всем файлам, использующим statedir. Если у вас есть “/var/run/frr” в качестве statedir по умолчанию, тогда он станет “/var/run/frr/<пространство имен>”

-o , --vrfdefaultname <name>

Задайте имя, используемое для *VRF по умолчанию* в командах командной строки и моделях YANG. Этот параметр должен быть одинаковым для всех запущенных демонов. По умолчанию используется имя “по умолчанию”.

-v, --version

Распечатать версию программы.

--command-log-always

Заставьте демона всегда записывать введенные команды в указанный файл журнала. Это также делает команду ***no log commands*** запрещенной. Включение этой функции рекомендуется, если вам необходимо отслеживать, что оператор делает на этом маршрутизаторе.

--log <stdout|syslog|file:/path/to/log/file>

При инициализации демона настройте журнал для перехода в стандартный вывод, системный журнал или в файл. Эти значения будут отображаться как часть демонстрационного запуска. Кроме того, они могут быть переопределены во время выполнения, если это необходимо, с помощью обычных команд журнала.

--log-level <emergencies|alerts|critical|errors|warnings|notifications|informational|debugging>

При инициализации демона разрешите указывать уровень журнала по умолчанию при запуске с одного из указанных уровней.

--tcli

Включите транзакционный режим командной строки.

--limit-fds <number>

Ограничьте количество файловых дескрипторов, которые будут использоваться внутри демонами FRR. По умолчанию демоны используют системное значение ulimit.

1.8.2.10 Поддержка загружаемых модулей

FRR поддерживает загрузку модулей расширения при запуске. Загрузка, перезагрузка или выгрузка модулей во время выполнения не поддерживается (пока). Чтобы загрузить модуль, используйте следующую опцию командной строки при запуске демона:

-M , --module <module:options>

Загрузите указанный модуль, при необходимости передав ему параметры. Если имя модуля содержит косую черту (/), предполагается, что это полный путь к загружаемому файлу. Если он не содержит косой черты, в каталоге /usr/lib/frr/modules выполняется поиск модуля с заданным именем; сначала с добавлением имени демона (например `zebra_mod`, для `mod`), затем без добавления имени демона.

Эта опция доступна для всех демонов, хотя некоторые демоны могут не иметь доступных для загрузки модулей.

1.8.2.11 Модуль SNMP

Если SNMP включен во время компиляции и установлен как часть пакета, `snmp` модуль может быть загружен для демонов Zebra, bgpd, ospfd, ospf6d и ripd.

Модуль игнорирует любые переданные ему параметры. Обратитесь к поддержке SNMP для получения информации о ее использовании.

1.8.2.12 **Модуль FPM**

Если FPM включен во время компиляции и установлен как часть пакета, fpm модуль может быть загружен для демона zebra. Это обеспечивает API диспетчера плоскости пересылки (“FPM”).

Модуль ожидает, что его аргументом будет либо Netlink или protobuf, указывающий используемую инкапсуляцию. Netlink используется по умолчанию и protobuf может быть недоступен, если модуль был собран без поддержки protobuf. Для получения дополнительной информации обратитесь к интерфейсу zebra FIB push.

1.8.2.13 **Интерфейсы виртуальных терминалов**

Интерфейс VTY – Virtual Terminal [он же TeletYpe] - это интерфейс командной строки (CLI) для взаимодействия пользователя с демоном маршрутизации.

1.8.2.13.1 Обзор VTY

VTY означает виртуальный телетайпный интерфейс. Это означает, что вы можете подключиться к демону через протокол telnet.

Чтобы включить интерфейс VTY, необходимо установить пароль VTY. Если нет пароля VTY, невозможно подключиться к интерфейсу VTY вообще.

```
% telnet localhost 2601
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.

Hello, this is |PACKAGE_NAME| (version |PACKAGE_VERSION|)
|COPYRIGHT_STR|

User Access Verification

Password: XXXXX
Router> ?
 enable . . . Turn on privileged commands
 exit . . . Exit current mode and down to previous mode
 help . . . Description of the interactive help system
 list . . . Print command list
 show . . . Show system inform

 wh. . . Display who is on a vty
Router> enable
Password: XXXXX
Router# configure terminal
Router(config)# interface eth0
Router(config-if)# ip address 10.0.0.1/8
Router(config-if)# ^Z
Router#
```

1.8.2.13.2 VTY Modes

Существует три основных режима VTY:

Существуют команды, которые могут быть ограничены определенными режимами VTY.

1.8.2.13.3 VTY View Model

Этот режим предназначен для доступа к CLI только для чтения. Можно выйти из режима, выйдя из системы или введя *включить*.

1.8.2.13.4 VTY Enable Model

Этот режим предназначен для доступа к CLI для чтения и записи. Можно выйти из режима, выйдя из системы или перейдя в режим просмотра.

1.8.2.13.5 VTY Other Modes

Эта страница предназначена для описания других режимов.

1.8.2.13.6 VTY CLI Commands

Команды, которые вы можете использовать в командной строке, описаны в следующих трех подразделах.

1.8.2.13.7 CLI Movement Commands

Эти команды используются для перемещения курсора командной строки. С С символ означает нажатие клавиши управления.

C-f / LEFT

Переместите вперед на один символ.

C-b / RIGHT

Переместите назад на один символ.

M-f

Переместитесь на одно слово вперед.

M-b

Переместитесь на одно слово назад.

C-a

Переместитесь в начало строки.

C-e

Переместитесь в конец строки.

1.8.2.13.8 Команды редактирования CLI

Эти команды используются для редактирования текста в строке. ССимвол означает нажатие клавиши управления.

C-h / DEL

Удалите символ перед точкой.

C-d

Удалите символ после точки.

M-d

Переслать уничтожающее слово.

C-w

Слово уничтожения в обратном направлении.

C-k

Завершите выполнение до конца строки.

C-u

Завершите строку с самого начала, стирая ввод.

C-t

Транспонировать символ.

1.8.2.13.9 Расширенные команды CLI

Существует несколько дополнительных команд CLI для завершения командной строки, insta-help и управления сеансами VTY.

C-c

Прерывание текущего ввода и переход к следующей строке.

C-z

Завершите текущий сеанс настройки и перейдите к верхнему узлу.

C-n / DOWN

Перейдите к следующей строке в буфере истории.

C-p / UP

Перейдите к предыдущей строке в буфере истории.

TAB

Используйте завершение командной строки, введя TAB.

?

Вы можете использовать справку командной строки, введя helpв начале строки. Ввод ? в любой точке строки покажет возможные завершения.

1.8.2.13.10 Действия канала

VTY поддерживает необязательные модификаторы в конце команд, которые выполняют постобработку при выводе команды или изменяют действие команд. Они не отображаются в ? TAB списках предложений или.

... | include REGEX

Фильтрует вывод предыдущей команды, включая только те строки, которые соответствуют расширенному регулярному выражению POSIX `REGEX`. Не заключайте регулярное выражение в кавычки.

Примеры:

```
frr# show ip bgp sum json | include remoteAs
"remoteAs":0,
"remoteAs":455,
"remoteAs":99,
```

```
frr# show run | include neigh.*[0-9]{2}\.0\.[2-4]\.[0-9]*
neighbor 10.0.2.106 remote-as 99
neighbor 10.0.2.107 remote-as 99
neighbor 10.0.2.108 remote-as 99
neighbor 10.0.2.109 remote-as 99
neighbor 10.0.2.110 remote-as 99
neighbor 10.0.3.111 remote-as 111
```

1.8.3 Расширенное ведение журнала

1.8.3.1 Пункты назначения

Местоположение вывода настраивается с помощью следующих подкоманд:

destination none

Отключите цель, сохранив ее оставшуюся конфигурацию.

destination syslog[supports-rfc5424]

Отправляйте сообщения журнала в стандартное место назначения системного журнала (`/dev/log`). При этом не используются библиотеки C `syslog()` функция, вместо записи непосредственно в `/dev/log` .

destination

Отправляйте сообщения журнала в журнал systemd.

destination <stdout|stderr|fd <(0-63)|envvar WORD>> [format FORMAT]

Отправляйте сообщения журнала в один из файловых дескрипторов демона. The `fd (0-63)` и `fd envvar WORD` варианты предназначены для работы с `command 3>somethingcommand {ENVVAR}>something` и спецификаторы перенаправления ввода-вывода bash.

Для этого могут использоваться только файловые дескрипторы, открытые во время запуска демона; предотвращается случайное неправильное использование файлового дескриптора, который был открыт самим FRR.

Использование FIFOs с этой опцией будет работать, но не поддерживается и может привести к зависанию или сбою демонов в зависимости от поведения читателя.

По умолчанию используется формат RFC5424, если не указан ни один.

destination filePATH[create [{user WORD|group WORD|mode PERMS}]|no create] [format FORMAT]

Войдите в обычный файл. Права доступа к файлам могут быть указаны, когда FRR создает сам файл.

По умолчанию используется формат RFC5424, если не указан ни один.

destination unix PATH[format FORMAT]

Подключитесь к сокету домена UNIX и отправляйте туда сообщения журнала.

Это приведет к автоматическому определению `SOCK_STREAMSOCK_SEQPACKETSOCK_DGRAM`, и соответствующим образом скорректировать поведение.

1.8.3.2 Опции

priority PRIORITY

Выберите минимальный приоритет сообщений для отправки этой цели. По умолчанию *отладка*.

facility FACILITY

Выберите средство системного журнала для сообщений по этой цели. По умолчанию *демон*. `log facility [FACILITY]` Команда не влияет на расширенные цели.

timestamp (0-9)

Установите желаемое количество цифр временной метки с точностью до секунды. Это действует только для целей форматов RFC5424 и journald; форматы RFC3164 и local-syslogd не поддерживают никаких субсекундных цифр.

timestamp local-time

Используйте часовой пояс локальной системы для временных меток, а не UTC (по умолчанию).

Форматы RFC5424 и journald содержат информацию о зоне (`z` или `+-NN:NN` суффикс в ISO8601). RFC3164 и local-syslogd не предлагают способа определения используемого часового пояса, необходимо позаботиться о том, чтобы этот параметр и приемник были настроены идентично, иначе временная метка в приемнике будет заменена.

structured-data <code-location|version|unique-id|error-category|format-args>

Выберите дополнительные данные ключа / значения, которые будут включены для форматов RFC5424 и journald. Подробности см. в следующем разделе. `unique-id` и `error-category` по умолчанию включены.

1.8.3.3 Структурированные данные

При использовании форматов RFC5424 или journaldr FRR может предоставлять дополнительные метаданные для сообщений журнала в виде пар ключ / значение. Таким образом можно добавить следующую информацию:

Переключить	группа 5424	5424 элемента (ов)	поле журнала	Содержание
всегда активен	location@50145	tid	TID	Идентификатор потока
всегда активен	location@50145	instance	FRR_INSTANCE	Номер нескольких экземпляров
unique-id	location@50145	id	FRR_ID	XXXXX-XXXXX уникальный идентификатор сообщения
error-category	location@50145	ec	FRR_EC	Целочисленный номер категории ошибки
code-location	location@50145	file	CODE_FILE	Имя файла исходного кода
code-location	location@50145	line	CODE_LINE	Номер строки исходного кода
code-location	location@50145	func	CODE_FUNC	Имя функции исходного кода
format-args	args@50145	argN	FRR_ARGn	Аргументы формата printf сообщения (n = 1..16)
version	origin	несколько	n/a	Информация о версии FRR (формат IETF)

Информация,добавляемая `version[origin enterpriseId="50145" software="FRRouting" swVersion="..."]` и одинаковая для всех сообщений журнала. (Следовательно, нет смысла включать в большинство сценариев.) 50145 - это номер предприятия FRRouting IANA.

Аварийные журналы / обратные трассировки не содержат никакой дополнительной информации, поскольку ее нельзя безопасно получить из обработчика сбоев. Однако все вышеуказанные пункты назначения будут доставлять аварийные журналы.

1.8.4 **BGP**

BGP расшифровывается как протокол пограничного шлюза. Последняя версия BGP - 4. BGP-4 является одним из протоколов внешнего шлюза и де-факто стандартным протоколом междоменной маршрутизации. BGP-4 описан в RFC 1771 и обновлен RFC 4271. RFC 2858 добавляет в BGP-4 поддержку многопротокольных протоколов.

1.8.4.1 **Запуск BGP**

Файл конфигурации bgpd по умолчанию bgpd.conf. bgpd выполняет поиск сначала в текущем каталоге, а затем в /etc/frr/bgpd.conf. Все команды bgpd должны быть настроены bbgpd.conf, когда встроенная конфигурация не используется.

Ниже описаны конкретные параметры вызова bgpd. Также могут быть указаны общие параметры (общие параметры вызова).

-p , --bgp_port <port>

Задайте номер порта протокола bgp. Если номер порта равен 0, это означает, что не прослушивать порт bgp.

-l, --listenon

Укажите конкретные IP-адреса для прослушивания bgpd, а не 0.0.0.0/ по умолчанию ::. Это может быть полезно для ограничения bgpd внутренним адресом или для запуска нескольких процессов bgpd на одном хосте. Можно указать несколько адресов.

В следующем примере bgpd запускается для прослушивания соединений по адресам 100.0.1.2 и fd00::2:2. В этом примере также используются опции -d (выполняется в режиме демона) и -f (используется определенный файл конфигурации), поскольку при использовании опции -l мы, вероятно, будем запускать несколько экземпляров bgpd, каждый с разными конфигурациями.

Обратите внимание, что этот параметр подразумевает параметр –no_kernel , и никакие изученные маршруты не будут установлены в ядро Linux.

 # /usr/lib/frr/bgpd -d -f /some-folder/bgpd.conf -l 100.0.1.2 -l fd00::2:2

-n, --no_kernel

Не устанавливайте изученные маршруты в ядро Linux. Этот параметр полезен для среды с отражателем маршрутов или если вы запускаете несколько процессов bgp в одном пространстве имен. Этот параметр отличается от параметра –no_zebra тем, что выполняется подключение по протоколу ZAPI.

Этот параметр также можно переключать во время выполнения с помощью [no] bgp no-ribкоманд в командной оболочке VTY.

Обратите внимание, что этот параметр будет сохраняться после сохранения конфигурации во время выполнения, если только он не будет отключен no bgp no-ribкомандой в командной строке VTY перед операцией записи конфигурации.

-S, --skip_runas

Пропустите обычный процесс проверки возможностей и изменения информации о пользователях и группах.

-e, --ecmp

Запустите BGP с ограниченными возможностями ecmp, которые отличаются от того, с которым был скомпилирован BGP. Указанное значение должно быть больше 0 и меньше или равно MULTIPATH_NUM, указанному при компиляции.

-Z, --no_zebra

Вообще не связывайтесь с zebra. Это отличается от опции –no_kernel тем, что мы даже не открываем соединение ZAPI с процессом zebra.

-s, --socket_size

При открытии tcp-соединений с нашими одноранговыми узлами задайте размер буфера отправки сокета, который ядро будет использовать для сокета одноранговых узлов. Эта опция действительно полезна только в очень больших масштабах. Следует провести эксперименты, чтобы увидеть, помогает ли это или нет в том масштабе, в котором вы работаете.

1.8.4.2 Менеджер меток

-I, --int_num

Задайте идентификатор zclient. Это требуется при использовании Zebra Label manager в режиме прокси.

1.8.4.3 Основные понятия

1.8.4.3.1 Автономные системы

Из RFC 1930:

AS - это подключенная группа из одного или нескольких IP-префиксов, управляемая одним или несколькими сетевыми операторами, которая имеет ЕДИНУЮ и ЧЕТКО ОПРЕДЕЛЕННУЮ политику маршрутизации.

С каждым AS связан идентификационный номер, называемый ASN. Это значение с двумя октетами в диапазоне от 1 до 65535. Номера AS с 64512 по 65535 определены как частные номера AS. Частные номера AS не должны рекламироваться в глобальном Интернете.

ASN является одним из важнейших элементов BGP. BGP - это протокол маршрутизации вектора расстояния, а платформа AS-Path предоставляет BGP метрику вектора расстояния и обнаружение циклов.

1.8.4.3.2 Семейства адресов

Многопротокольные расширения позволяют BGP передавать информацию о маршрутизации для нескольких протоколов сетевого уровня. BGP поддерживает идентификатор семейства адресов (AFI) для IPv4 и IPv6. Также предоставляется поддержка нескольких наборов информации для каждого AFI с помощью идентификатора семейства последующих адресов BGP (SAFI). FRR поддерживает SAFI для одноадресной информации, помеченной информации (RFC 3107 и RFC 8277) и информации VPN уровня 3 (RFC 4364 и RFC 4659).

1.8.4.3.3 Выбор маршрута

Процесс выбора маршрута, используемый реализацией FRR BGP, использует следующий критерий принятия решения, начиная с верхней части списка и двигаясь вниз, пока не будет использован один из факторов.

1. **Проверка веса**

Отдавайте предпочтение маршрутам с более высоким локальным весом маршрутам с более низким весом.

2. **Проверка местных предпочтений**

Предпочитайте более высокие локальные предпочтительные маршруты более низким.

Если bgp bestpath aigp этот параметр включен и оба сравниваемых пути имеют атрибут AIGP, BGP использует разрыв связей AIGP, если только оба пути не имеют атрибута метрики AIGP. Это означает, что атрибут AIGP не оценивается в процессе выбора наилучшего пути между двумя путями, когда один путь не имеет атрибута AIGP.

3. **Проверка локального маршрута**

Предпочитайте локальные маршруты (статические, агрегатные, перераспределенные) полученным маршрутам.

4. **Проверка длины пути AS**

Предпочитайте AS_PATH с кратчайшим числом переходов.

5. **Origin check**

Отдавайте предпочтение маршруту с наименьшим исходным типом. То есть предпочесть исходные маршруты IGP EGP, а не неполным маршрутам.

6. **MED проверка**

Если маршруты с MED были получены от одной и той же AS, предпочтение отдается маршруту с наименьшим MED. :ref:`bgp-med` .

7. **Внешняя проверка**

Отдавать предпочтение маршруту, полученному от внешнего одноранговой узла eBGP, а не маршрутам, полученным от других типов одноранговых узлов.

8. **Проверка стоимости IGP**

Отдайте предпочтение маршруту с более низкой стоимостью IGP.

9. **Multi-path check**

Если многопутевое соединение включено, проверьте, можно ли считать маршруты, еще не различающиеся по предпочтениям, равными. Если установлен параметр :clicmd:`bgp bestpath as-path multipath-relax` , все такие маршруты считаются равными, в противном случае равными считаются маршруты, полученные через iBGP с идентичными AS_PATH, или маршруты, полученные от соседей eBGP в одной и той же AS.

10. **Already-selected external check**

Если оба маршрута были получены от одноранговых узлов eBGP, предпочтение отдается уже выбранному маршруту. Обратите внимание, что эта проверка не применяется, если настроен параметр :clicmd:`bgp bestpath compare-routerid` . Эта проверка может предотвратить некоторые случаи осцилляции.

11. **Проверка идентификатора маршрутизатора**

Отдайте предпочтение маршруту с наименьшим идентификатором маршрутизатора. Если маршрут имеет атрибут ORIGINATOR_ID через отражение iBGP, то используется этот идентификатор маршрутизатора, в противном случае используется идентификатор маршрутизатора одноранговой узла, от которого был получен маршрут.

12. **Проверка длины Cluster-List**

Используется маршрут с наименьшей длиной списка кластеров. Список кластеров отражает путь отражения iBGP, по которому прошел маршрут.

13. **Адрес пира**

Предпочитайте маршрут, полученный от узла с более высоким адресом транспортного уровня, в качестве последнего средства разрешения конфликтов.

1.8.4.3.4 Согласование возможностей

При добавлении функции обмена информацией о маршрутизации IPv6 в BGP. Были некоторые предложения. :abbr:`IETF (Internet Engineering Task Force)` :abbr:`IDR (междоменная маршрутизация)` приняла предложение под названием Multiprotocol Extension for BGP. Спецификация описана в RFC 2283. Протокол не определяет новые протоколы. Он определяет

новые атрибуты для существующего BGP. Когда он используется для обмена информацией о маршрутизации IPv6, он называется BGP-4+. Когда он используется для обмена информацией о многоадресной маршрутизации, он называется MBGP.

bgpd поддерживает многопротокольное расширение для BGP. Таким образом, если удаленный узел поддерживает протокол, bgpd может обмениваться информацией IPv6 и/или многоадресной маршрутизации.

Традиционный BGP не имел возможности определять возможности удаленного партнера, например, может ли он обрабатывать типы префиксов, отличные от одноадресных маршрутов IPv4. Это было большой проблемой при использовании Multiprotocol Extension для BGP в действующей сети. RFC 2842 принял функцию, называемую Capability Negotiation. bgpd использует это согласование возможностей для определения возможностей удаленного узла. Если одноранговый узел настроен только как одноадресный сосед IPv4, bgpd не отправляет эти пакеты согласования возможностей (по крайней мере, если другие дополнительные функции BGP не требуют согласования возможностей).

По умолчанию FRR запускает пиринг с минимальными общими возможностями для обеих сторон. Например, если локальный маршрутизатор поддерживает одноадресную и многоадресную рассылку, а удаленный маршрутизатор поддерживает только одноадресную передачу, локальный маршрутизатор установит соединение только с одноадресной передачей. Если общих возможностей нет, FRR отправляет ошибку Unsupported Capability, а затем сбрасывает соединение.

1.8.4.4 Конфигурация маршрутизатора BGP

1.8.4.4.1 ASN и идентификатор маршрутизатора

Прежде всего, вы должны настроить маршрутизатор BGP с router bgp ASNпомощью команды. Номер AS является идентификатором автономной системы. Протокол BGP использует номер AS для определения, является ли соединение BGP внутренним или внешним.

router bgp ASN

Включите процесс протокола BGP с указанным ASN. После этой инструкции вы можете вводить любые команды BGP.

bgp router-idA.B.C.D

Эта команда определяет идентификатор маршрутизатора. Если bgpd подключается к zebra, он получает информацию об интерфейсе и адресе. В этом случае значение идентификатора маршрутизатора по умолчанию выбирается как самый большой IP-адрес интерфейсов. Когда маршрутизатор zebra не включен, bgpd не может получить информацию об интерфейсе, поэтому для идентификатора маршрутизатора установлено значение 0.0.0.0. Поэтому, пожалуйста, установите идентификатор маршрутизатора вручную.

1.8.4.4.2 Несколько автономных систем

Реализация BGP в FRR способна запускать несколько автономных систем одновременно. Каждый настроенный AS соответствует виртуальной маршрутизации и пересылке. В прошлом, чтобы получить ту же функциональность, сетевому администратору приходилось запускать новый процесс bgpd; использование VRFs позволяет обрабатывать несколько автономных систем в одном процессе.

При использовании нескольких автономных систем все блоки конфигурации маршрутизатора после первого должны указывать VRF в качестве цели выбора маршрута BGP. Этот VRF должен быть уникальным по отношению ко всем другим VRF, используемым для той

же цели, т. Е. Две разные автономные системы не могут использовать один и тот же VRF. Тем не менее, то же самое, что можно использовать с разными VRFS.

Примечание

Разделенный характер VRFs позволяет подключать один процесс bgpd к самому себе на одной машине. Обратите внимание, что это может быть полностью выполнено в BGP без соответствующего VRF в ядре или Zebra, что позволяет использовать некоторые практические варианты использования, такие как отражатели маршрутов и серверы маршрутов.

Настройка дополнительных автономных систем или маршрутизатора, который настроен на определенный VRF, выполняется с помощью следующей команды:

router bgp ASN vrf VRFNAME

VRFNAME сопоставляется с VRFS, настроенной в ядре. Если vrf VRFNAMEне указано, процесс протокола BGP относится к VRF по умолчанию.

Пример конфигурации с несколькими автономными системами может выглядеть следующим образом:

```
router bgp 1
neighbor  10.0.0.1 remote-as 20
 neighbor  10.0.0.2 remote-as 30
!
router bgp 2 vrf blue
 neighbor  10.0.0.3 remote-as 40
 neighbor  10.0.0.4 remote-as 50
!
router bgp 3 vrf red
 neighbor  10.0.0.5 remote-as 60
 neighbor  10.0.0.6 remote-as 70
...
```

1.8.4.4.3 Просмотры

В дополнение к поддержке нескольких автономных систем, реализация FRR BGP также поддерживает представления.

Представления BGP почти такие же, как и обычные процессы BGP, за исключением того, что маршруты, выбранные BGP, не устанавливаются в таблицу маршрутизации ядра. Каждое представление BGP предоставляет независимый набор информации о маршрутизации, которая распространяется только через BGP. Может поддерживаться несколько представлений, и информация о представлении BGP всегда независима от других протоколов маршрутизации и маршрутов Zebra / kernel. Представления BGP используют экземпляр ядра (т. Е. VRF по умолчанию) для связи с одноранговыми узлами.

router bgpAS-NUMBER viewNAME

Создайте новое представление BGP. Вы можете использовать произвольное слово для NAME. Маршруты, выбранные представлением, не устанавливаются в таблицу маршрутизации ядра.

С помощью этой команды вы можете настроить сервер маршрутизации, как показано ниже.

```
!
! bgp 1 view 1
 neighbor  10.0.0.1 remote-as 2
 neighbor  10.0.0.2 remote-as 3
!
router bgp 2 view 2
 neighbor  10.0.0.3 remote-as 4
neighbor  10.0.0.4 remote-as 5
```

show [ip] bgp

Отображение таблицы маршрутизации в представлении BGP `NAME`.

1.8.4.4.4 Выбор маршрута

bestpath

Эта команда указывает, что длина наборов и последовательностей путей объединения должна учитываться в процессе принятия решения о наилучшем пути BGP.

as-path

Эта команда указывает, что процесс принятия решения BGP должен рассматривать пути равной длины AS_PATH, которые являются кандидатами для многолучевого вычисления. Без ручки весь AS_PATH должен совпадать для многолучевого вычисления.

bgp bestpath compare-routerid

Убедитесь, что при сравнении маршрутов, где оба равны по большинству показателей, включая local-pref, длину AS_PATH, стоимость IGP, MED, связь нарушена на основе идентификатора маршрутизатора.

Если эта опция включена, то уже выбранная проверка, в которой предпочтительными являются уже выбранные маршруты eBGP, пропускается.

Если у маршрута есть атрибут ORIGINATOR_ID, потому что он был отражен, будет использоваться этот ORIGINATOR_ID. В противном случае будет использоваться идентификатор маршрутизатора узла, от которого был получен маршрут.

Преимущество этого заключается в том, что выбор маршрута (на данном этапе) будет более детерминированным. Недостатком является то, что несколько или даже один маршрутизатор с наименьшим идентификатором могут привлекать весь трафик по другим равным путям из-за этой проверки. Это может увеличить вероятность колебаний MED или IGP, если не были приняты другие меры для их предотвращения. Точное поведение будет зависеть от топологии iBGP и отражения.

bgp bestpath peer-type multipath-relax

Эта команда указывает, что процесс принятия решения BGP должен учитывать пути от всех одноранговых узлов для многолучевых вычислений. Если эта опция включена, пути, полученные от любого из соседей eBGP, iBGP или конфедерации, будут многолучевыми, если в противном случае они считаются равными по стоимости.

bgp bestpath aigp

Используйте команду bgp bestpath aigp для оценки атрибута AIGP в процессе выбора наилучшего пути между двумя путями, имеющими атрибут AIGP.

Когда bgp bestpath aigp отключен, BGP не использует правила AIGP, нарушающие связь, если пути не имеют атрибута AIGP.

По умолчанию отключен.

maximum-paths (1-128)

Задает значение максимального пути, используемое для вычислений ecmp для этого экземпляра bgp в EBGP. Максимальное указанное значение, 128, может быть ограничено cli ecmp для bgp или если демон был скомпилирован с меньшим значением ecmp. Это значение также может быть установлено в одноадресной / одноадресной передаче ipv4 / ipv6, чтобы влиять только на эти конкретные afi / safi.

maximum-paths ibgp (1-128) [equal-cluster-length]

Задает значение максимального пути, используемое для вычислений ecmp для этого экземпляра bgp в IBGP. Максимальное указанное значение, 128, может быть ограничено cli ecmp для bgp или если демон был скомпилирован с меньшим значением ecmp. Это значение также может быть установлено в одноадресной / одноадресной передаче ipv4 / ipv6, чтобы влиять только на эти конкретные afi / safi.

1.8.4.4.5 Административные показатели расстояния

distance bgp (1-255) (1-255) (1-255)

Эта команда изменяет значение расстояния для BGP. Аргументами являются значения расстояния для внешних маршрутов, внутренних маршрутов и локальных маршрутов соответственно.

distance (1-255) A.B.C.D/M

distance (1-255) A.B.C.D/M WORD

Задает административное расстояние для определенного маршрута.

1.8.4.4.6 Требуется политика для EBGP

bgp ebgp-requires-policy

Эта команда требует, чтобы входящие и исходящие фильтры применялись для сеансов eBGP в соответствии с требованиями RFC-8212. Без входящего фильтра никакие маршруты не будут приниматься. Без фильтра исходящих маршруты не будут объявлены.

Это включено по умолчанию для традиционной конфигурации и отключено по умолчанию для конфигурации центра обработки данных.

При включении/ отключении этой опции НЕОБХОДИМО очистить сеанс.

Если фильтр входящих или исходящих сообщений отсутствует, вы увидите “(Политика)” войдите bshow bgp summary:

```
exit1# show bgp summary

IPv4 Unicast Summary (VRF default):
BGP 10.10.10.1, local AS number 65001 vrf-id 0
BGP 4
```

```
RIB 7, using 1344 bytes of memory
2 Peers 2, using 43 KiB of memory

Neighbor V AS MsgRcvd MsgSent TblVer InQ OutQ Up/Down State/PfxRcd PfxSnt Desc
192.168.0.2     4      65002        8        10        0    0    0  00:03:09            5 (Policy) N/A
fe80:1::2222    4      65002        9        11        0    0    0  00:03:09 (Policy) (Policy) N/A
```

Кроме того, *показать соседнего bgp* команда будет указывать в *Для семейства адресов:* заблокируйте это:

```
exit1# show bgp neighbor
...
For address family: IPv4 Unicast
 Update group 1, subgroup 1
 Packet Queue length 0
 Inbound soft reconfiguration allowed
 Community attribute sent to this neighbor(all)
 Inbound updates discarded due to missing policy
 Outbound updates discarded due to missing policy
 0 accepted prefixes
```

1.8.4.4.7 Отклонять маршруты с типами AS_SET или AS_CONFED_SET

bgp reject-as-sets

Эта команда позволяет отклонять входящие и исходящие маршруты, имеющие тип AS_SET или AS_CONFED_SET.

1.8.4.4.8 Подавление повторяющихся обновлений

bgp suppress-duplicates

Например, маршрутизаторы BGP могут генерировать несколько идентичных объявлений с пустыми атрибутами сообщества, если они удалены при выходе. Это нежелательное поведение. Подавлять повторяющиеся обновления, если маршрут фактически не изменился. По умолчанию: включено.

1.8.4.4.9 Отправить уведомление о прекращении жесткого сброса для административного сброса

bgp hard-administrative-reset

Отправка уведомления о прекращении жесткого сброса для событий ‘административного сброса’.

Когда отключено, и между одноранговыми узлами осуществляется обмен уведомлениями о плавном перезапуске, применяются процедуры плавного перезапуска, и маршруты будут сохранены.

По умолчанию включено.

1.8.4.4.10 Отключить проверку, подключен ли nexthop к сеансам EBGP

bgp disable-ebgp-connected-route-check

Эта команда используется для отключения процесса проверки соединения для сеансов однорангового соединения EBGP, которые доступны за один переход, но настроены на петлевой интерфейс или иным образом настроены с IP-адресом, не подключенным напрямую.

1.8.4.4.11 Route Flap Dampening

bgp dampening(1-45) (1-20000) (1-50000) (1-255)

Эта команда включает flap маршрута BGP и задает параметры демпфирования.

half-life

Время полураспада

reuse-threshold

Значение для начала повторного использования маршрута

suppress-threshold

Значение для начала подавления маршрута

max-suppress

Максимальная продолжительность для подавления стабильного маршрута

Алгоритм демпфирования маршрута с закрылками совместим с RFC 2439. В настоящее время использование этой команды не рекомендуется.

На данный момент демпфирование с закрытием маршрута не работает для VRF и работает только для одноадресной и многоадресной рассылки IPv4.

1.8.4.4.12 Multi-Exit Discriminator

Атрибут BGP MED обладает свойствами, которые могут вызвать незначительные проблемы с конвергенцией в BGP. Эти свойства и проблемы оказались трудными для понимания, по крайней мере исторически, и, возможно, до сих пор не получили широкого понимания. Следующие попытки собрать воедино и представить то, что известно о MED, чтобы помочь операторам и пользователям FRR в проектировании и настройке своих сетей.

Атрибут BGP MED предназначен для того, чтобы позволить одному AS указывать свои предпочтения для точек входа в другой AS. Атрибут MED не будет передаваться на другой AS принимающим AS - он ‘нетранзитивный’ в смысле BGP.

Например, если AS X и AS Y имеют 2 разные точки пиринга BGP, то AS X может установить среднее значение 100 для маршрутов, объявленных в одном, и среднее значение 200 для другого. Когда AS Y выбирает между равными в остальном маршрутами к или через AS X, AS Y должен предпочесть выбрать путь через более низкий MED, равный 100, с AS X. Настройка MED позволяет AS влиять на маршрутизацию, проводимую к нему в пределах другого, соседнего AS.

При таком использовании MED на самом деле не имеет смысла сравнивать значение MED на маршрутах, где отличается следующий AS на путях. Например, если у AS Y также был маршрут для некоторого назначения через AS Z в дополнение к маршрутам из AS X, и AS Z также установил MEDдля AS Y не имело бы смысла сравнивать средние значения AS Z с значениями AS X. Значения MED были установлены разными администраторами с разными системами отсчета.

Следовательно, поведение BGP по умолчанию заключается в том, чтобы не сравнивать значения MED по маршрутам, полученным от разных соседних ASE. В FRR это делается путем сравнения соседних, крайних слева, как в полученных AS_PATHs маршрутов, и сравнения MED только в том случае, если они совпадают.

К сожалению, такое поведение MED, которое иногда сравнивается по маршрутам, а иногда нет, в зависимости от свойств этих других маршрутов, означает, что MED может привести к тому, что порядок предпочтения по всем маршрутам не будет определен. То есть, учитывая маршруты A, B и C, если A предпочтительнее B, а B предпочтительнее C, то четко определенный

порядок должен означать, что предпочтение является транзитивным (в смысле порядков 1) и что A будет предпочтительнее C.

Однако, когда задействован MED, это не обязательно так. С MED возможно, что C на самом деле предпочтительнее, чем A. Таким образом, A предпочтительнее B, B предпочтительнее C, но C предпочтительнее A. Это может быть правдой даже там, где BGP определяет детерминированный "наиболее предпочтительный" маршрут из полного набора A, B, C. С MED для любого заданного набора маршрутов может существовать детерминированно предпочтительный маршрут, но не обязательно должен быть какой-либо способ упорядочить их в любом порядке предпочтения. При неизмененном MED порядок предпочтений маршрутов буквально становится неопределенным.

То, что MED может вызывать нетранзитивные предпочтения по маршрутам, может вызвать проблемы. Во-первых, это может восприниматься как локальная ошибка таблицы маршрутизации в динамиках; во-вторых, и это более серьезно, это может вызвать нестабильность маршрутизации в топологиях iBGP, где наборы динамиков постоянно колеблются между разными путями.

Первая проблема возникает из-за того, как спикеры часто реализуют решения о маршрутизации. Хотя BGP определяет процесс выбора, который будет детерминированно выбирать тот же маршрут, что и лучший для любого данного динамика, даже для MED, этот процесс требует совместной оценки всех маршрутов. По соображениям производительности и простоты реализации многие реализации вместо этого оценивают предпочтения маршрута попарно. Учитывая, что при использовании MED нет четко определенного порядка, наилучший маршрут, который будет выбран, зависит от деталей реализации, таких как порядок хранения маршрутов. Это может быть (локально) недетерминированным, например,: это может быть порядок, в котором были получены маршруты.

Этот индетерминизм может считаться нежелательным, хотя он не должен вызывать проблем. Это может означать, что воспринимается дополнительный отток маршрутизации, поскольку иногда в ответ на какое-либо событие может быть произведено больше обновлений, чем в другое время.

Эта первая проблема может быть исправлена с помощью более детерминированного выбора маршрута, который гарантирует, что маршруты упорядочиваются соседними AS во время выбора. bgp deterministic-med. Это может уменьшить количество обновлений по мере получения маршрутов и в некоторых случаях может уменьшить отток маршрутизации. Тем не менее, он может в равной степени детерминированно создавать максимально возможный набор обновлений в ответ на наиболее распространенную последовательность полученных обновлений.

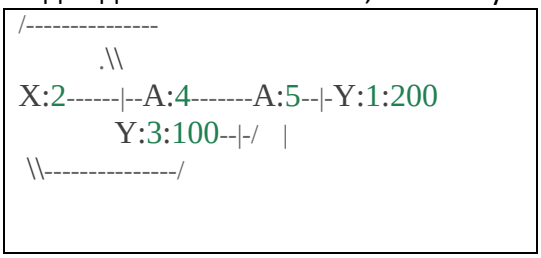
Детерминированный порядок вычисления, как правило, подразумевает дополнительные накладные расходы на сортировку по любому набору из n маршрутов к месту назначения. Реализация детерминированного MED в FRR масштабируется значительно хуже, чем большинство алгоритмов сортировки в настоящее время, с количеством путей к заданному месту назначения. Это число часто достаточно мало, чтобы не вызывать никаких проблем, но там, где путей много, детерминированное сравнение может быстро стать все более дорогостоящим с точки зрения процессора.

Детерминированная локальная оценка, однако, не может решить вторую, более серьезную проблему MED. Которая заключается в том, что нетранзитивное предпочтение маршрутов MED может привести к нестабильности маршрутизации или колебаниям между несколькими динамиками в топологиях iBGP. Это может произойти с полномешевым iBGP, но особенно проблематично в топологиях iBGP с неполной ячеей, которые еще больше уменьшают информацию о маршрутизации, известную каждому говорящему. В основном это было

задокументировано с помощью топологий iBGP с отражением маршрута. Однако любые технологии сокрытия маршрутов потенциально могут также усугубить колебания с MED.

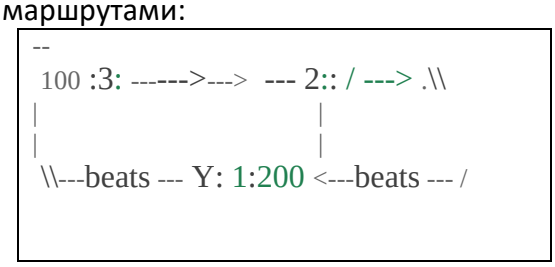
Эта вторая проблема возникает, когда у каждого говорящего есть только подмножество маршрутов, и в предпочтениях между различными комбинациями маршрутов есть циклы - как позволяет неопределенный порядок предпочтений MED - и маршруты распределены таким образом, что заставляют говорящих BGP ‘преследовать’ эти циклы. Это может произойти, даже если все динамики используют детерминированный порядок оценки при выборе маршрута.

Например, динамик 4 в КАЧЕСТВЕ A может получать маршрут от динамика 2 в ВИДЕ X, а от динамика 3 в ВИДЕ Y; в то время как динамик 5 в КАЧЕСТВЕ A может получать этот маршрут от динамика 1 в ВИДЕ Y. ПОСКОЛЬКУ Y может установить среднее значение 200 для динамика 1 и 100 для динамика3. То есть, используя ASN: ID: MED для обозначения динамиков:



Предполагая, что все остальные показатели равны (AS_PATH, ORIGIN, 0 IGP-затрат), затем , основываясь на процессе принятия решения RFC4271, спикер 4 выберет X: 2 вместо Y: 3: 100 на основе нижнего идентификатора 2. Спикер 4 объявляет X: 2 спикеру 5. Спикер 5 продолжитпредпочитайте Y: 1:200 на основе идентификатора и сообщите об этом спикеру 4. Теперь динамик 4 будет иметь полный набор маршрутов, и Y: 1: 200, который он получает от 5, будет превосходить X: 2, но когда динамик 4 сравнивает Y: 1: 200 с Y: 3: 100 , проверка MED теперь становится активной при совпадении ASES, и теперь Y: 3: предпочтительно 100. Поэтому докладчик 4 теперь объявляет Y: 3:100 до 5, который также соглашается с тем, что Y: 3:100 предпочтительнее, чем Y: 1: 200, и поэтому выводит последний маршрут из 4. Динамик 4 теперь имеет только X: 2 и Y: 3: 100, а X: 2 превосходит Y: 3: 100, и поэтому динамик 4 неявно обновляет свой маршрут к динамику 5 до X: 2. Динамик 5 видит, что Y: 1: 200 превосходит X: 2 на основе идентификатора, и объявляет громкоговорителю 4 Y: 1:200, и цикл продолжается.

Основной причиной является отсутствие четкого порядка предпочтений, вызванного тем, как MED иногда сравнивается, а иногда нет, что приводит к этому циклу в предпочтениях между маршрутами:



Этого конкретного типа колебаний в топологиях iBGP с полной ячеей можно избежать, если говорящие предпочитают уже выбранные внешние маршруты, а не предпочитают обновлять маршрут на основе метрики post-MED (например, router-ID) за счет недетерминированного процесса выбора. FRR реализует это, как и многие другие реализации, при условии, что оно не переопределяется установкой bgp bestpath compare-routerid, и смотрите также Выбор маршрута.

Однако с помощью iBGP route-reflection возможны более сложные и коварные циклы колебаний, избежать которых не так-то просто. Они были задокументированы в разных местах. См., например

bgp-route-osci-cond

stable-flexible-ibgp

ibgp-correctness

для конкретных примеров и дополнительных ссылок.

На момент написания этой статьи существует НЕТ известный способ использования MED по его первоначальному назначению; и сокращение информации о маршрутизации в топологиях iBGP; и обязательно избегайте проблем с нестабильностью MED из-за нетранзитивных предпочтений маршрутизации, которые он может вызвать; в общем, в произвольных сетях.

Могут существовать специфические для топологии iBGP способы снижения рисков нестабильности даже при использовании MED, например: путем ограничения топологии отражения и настройки затрат IGP между кластерами route-reflector, см. RFC 3345. В ближайшем будущем расширение Add-Path для BGP может также решить проблему колебаний MED, в то же время позволяя использовать MED по назначению, распределяя “наилучшие пути для каждого соседа КАК”. Это было бы связано с распределением по меньшей мере такого же количества маршрутов для всех динамиков, как и в полномешевом iBGP, если не больше, при этом накладывая аналогичные накладные расходы на процессор, как функция “детерминированного MED” на каждом отражателе добавленного пути.

В более общем плане, проблем с нестабильностью, которые MED может создавать в более сложных топологиях iBGP с неполной сеткой, можно избежать либо путем:

Настройка bgp always-compare-medоднако это позволяет сравнивать MED между значениями, установленными разными соседними ASE, что само по себе может не давать согласованных желаемых результатов.

Эффективное игнорирование MED путем установки MED на одно и то же значение (например: 0), используемое set metric METRICна всех принятых маршрутах, в сочетании с настройкой bgp always-compare-medна всех динамиках. Это самый простой и эффективный способ избежать проблем с колебаниями MED, когда AS счастлив не позволять соседям вводить эту проблемную метрику.

Поскольку MED оценивается после проверки длины AS_PATH, другим возможным использованием MED является управление внутри AS маршрутами с равной длиной AS_PATH, как продолжение последнего случая выше. Поскольку MED оценивается перед метрикой IGP, это может позволить реализовать маршрутизацию для отправки трафика на предпочтительные передачи с соседями, а не на ближайшую передачу в соответствии с метрикой IGP.

Обратите внимание, что даже если будут приняты меры для решения проблем нетранзитивности MED, могут быть возможны и другие колебания. Например, стоимость IGP, если топологии iBGP и IGP противоречат друг другу - см. Пример в статье Flavel и Roughan выше. Отсюда и указание, что топология iBGP должна соответствовать топологии IGP.

bgp deterministic-med

Выполняйте выбор маршрута таким образом, чтобы локально получать детерминированные ответы, даже несмотря на MED и отсутствие четко определенного порядка предпочтений, который он может вызвать на маршрутах. Без этой опции предпочтительный маршрут с MED может определяться в основном порядком, в котором были получены маршруты.

Установка этого параметра приведет к снижению производительности, что может быть заметно при наличии большого количества маршрутов для каждого пункта назначения. В настоящее время в FRR это реализовано таким образом, что плохо масштабируется по мере увеличения количества маршрутов на пункт назначения.

По умолчанию этот параметр не установлен.

Обратите внимание, что существуют и другие источники неопределенности в процессе выбора маршрута, в частности, предпочтение более старых и уже выбранных маршрутов от одноранговых узлов eBGP, Выбор маршрута.

bgp always-compare-med

Всегда сравнивайте MED на маршрутах, даже если они были получены от разных соседних ASE. Установка этого параметра делает порядок предпочтения маршрутов более определенным и должна устранить колебания, вызванные MED.

При использовании этой опции также может быть желательно set metric METRIC установить значение MED равным 0 для маршрутов, полученных от внешних соседей.

Эта опция может использоваться вместе с set metric METRICиспользованием MED в качестве метрики внутри AS для направления маршрутов AS_PATH равной длины, например, к желаемым точкам выхода.

1.8.4.4.13 Graceful restart

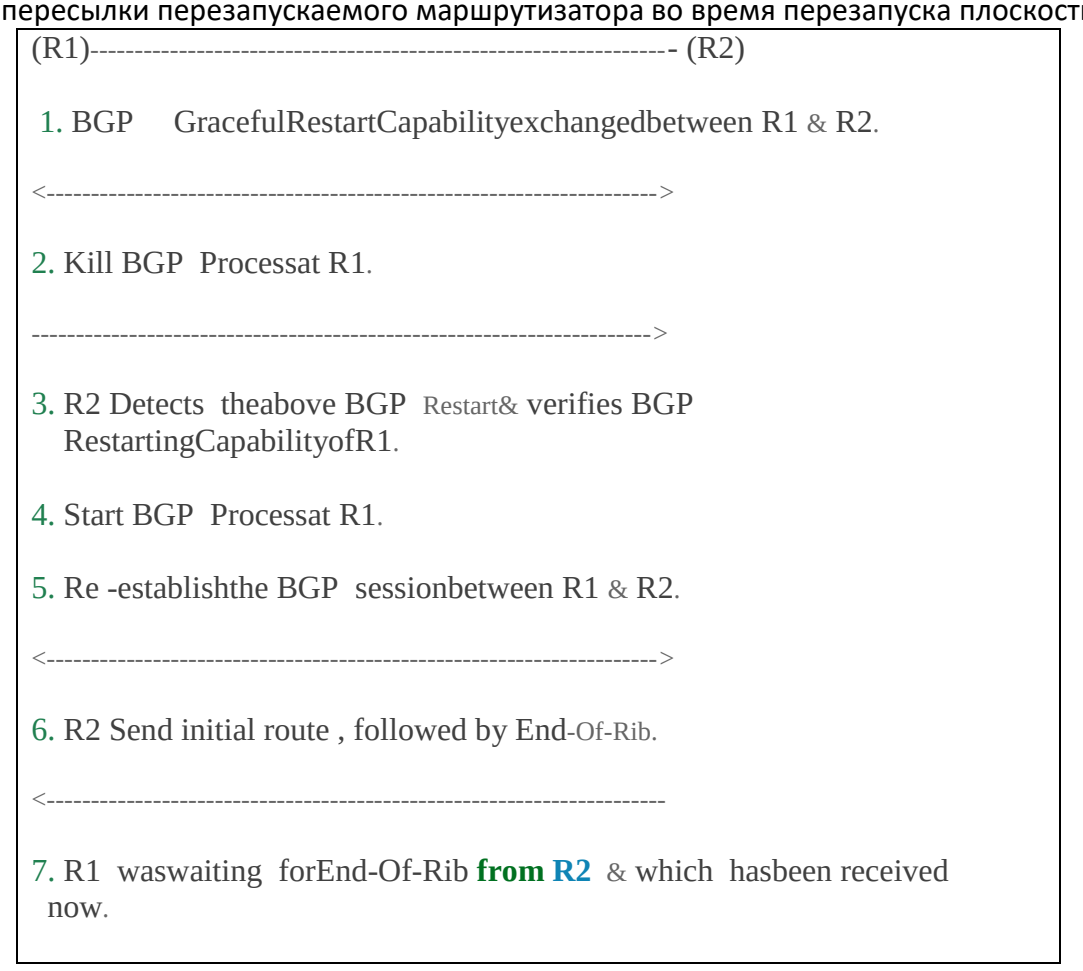
Функциональность BGP graceful restart, определенная в RFC-4724, определяет механизмы, которые позволяют BGP speaker продолжать пересылать пакеты данных по известным маршрутам, пока восстанавливается информация протокола маршрутизации.

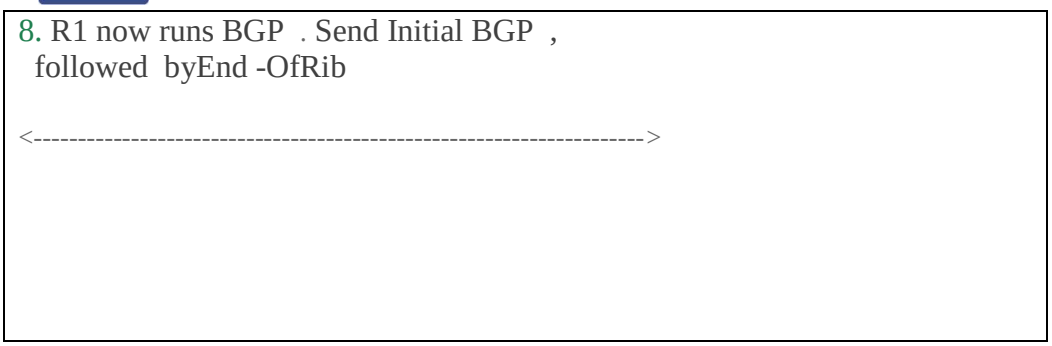
Обычно, когда BGP на маршрутизаторе перезапускается, все одноранговые узлы BGP обнаруживают, что сеанс прервался, а затем возобновился. Этот переход “вниз / вверх” приводит к “отклонению маршрутизации” и вызывает повторное вычисление маршрута BGP, генерацию обновлений маршрутизации BGP и ненужный отток в таблицах пересылки.

Следующие функциональные возможности предоставляются с помощью graceful restart:

Эта функция позволяет перезапускающему маршрутизатору указывать вспомогательному узлу маршруты, которые он может сохранить в своей плоскости пересылки во время перезапуска плоскости управления, отправляя возможность плавного перезапуска в ОТКРЫТОМ сообщении, отправленном во время установления сеанса.

Эта функция позволяет маршрутизатору сообщать всем другим одноранговым узлам маршруты, полученные от перезапускаемого маршрутизатора, которые сохраняются в плоскости пересылки перезапускаемого маршрутизатора во время перезапуска плоскости управления.





1.8.4.4.13.1 BGP-GR Preserve-Forwarding State

Открытое сообщение BGP, несущее дополнительные возможности для плавного перезапуска, имеет 8-разрядные “флаги для семейства адресов” для заданных AFI и SAFI. Это поле содержит битовые флаги, относящиеся к маршрутам, которые были объявлены с заданными AFI и SAFI.

Старший значащий бит определяется как бит состояния пересылки (F), который может использоваться для указания, действительно ли состояние пересылки для маршрутов, которые были объявлены с заданными AFI и SAFI, было сохранено во время предыдущего перезапуска BGP. При установке (значение 1) бит указывает, что состояние пересылки было сохранено. Оставшиеся биты зарезервированы и ДОЛЖНЫ быть установлены отправителем на ноль и проигнорированы получателем.

bgp graceful-restart preserve-fw-state

FRR дает нам возможность включить / отключить флаг “F” с помощью этой конкретной команды vty. Однако у него нет возможности включать / отключать этот флаг только для определенных AFI / SAFI, т.Е. Когда используется эта команда, она применяется ко всем поддерживаемым комбинациям AFI / SAFI для этого узла.

1.8.4.4.13.2 End-of-RIB (EOR) message

Сообщение ОБ ОБНОВЛЕНИИ с недоступной информацией о достижимости сетевого уровня (NLRI) и пустым удаленным NLRI указывается в качестве маркера конца ребра, который может использоваться динамиком BGP для указания своему одноранговому узлу завершения первоначального обновления маршрутизации после установления сеанса.

Для семейства одноадресных адресов IPv4 маркер конца строки представляет собой сообщение об ОБНОВЛЕНИИ минимальной длины. Для любого другого семейства адресов это сообщение об ОБНОВЛЕНИИ, которое содержит только атрибут MP_UNREACH_NLRI без отозванных маршрутов для этого <AFI, SAFI>.

Хотя маркер конца ребра указан для плавного перезапуска BGP, отмечается, что генерация такого маркера по завершении первоначального обновления была бы полезна для конвергенции маршрутизации в целом, и поэтому рекомендуется использовать эту практику.

1.8.4.4.13.3 Таймер отсрочки выбора маршрута

Указывает время, на которое перезапускаемый маршрутизатор откладывает процесс выбора маршрута после перезапуска.

Перезапуск маршрутизатора: использование таймера отсрочки выбора маршрута указано в https://tools.ietf.org/html/rfc4724#section-4.1

Как только сеанс между перезапускающим динамиком и принимающим динамиком будет восстановлен, перезапускающий динамик будет получать и обрабатывать сообщения BGP от своих одноранговых узлов.

Однако он ДОЛЖЕН отложить выбор маршрута для семейства адресов до тех пор, пока он либо.

1. Получает маркер конца ребра от всех своих одноранговых узлов (исключая те, у которых в полученной возможности установлен бит “Состояние перезапуска”, и исключая те, которые не рекламируют возможность плавного перезапуска).
2. Тайм-аут Selection_Deferral_Timer.

bgp graceful-restart select-defer-time (0-3600)

Эта команда установит время отсрочки на указанное значение.

bgp graceful-restart rib-stale-time (1-3600)

Эта команда устанавливает время, в течение которого устаревшие маршруты хранятся в RIB.

bgp graceful-restart restart-time (0-4095)

Установите время ожидания удаления устаревших маршрутов до получения сообщения BGP open.

При использовании с возможностью долговременного плавного перезапуска рекомендуется установить этот таймер на 0 и управлять устаревшими маршрутами с bgp long-lived-graceful-restart stale-time помощью.

Значение по умолчанию равно 120.

bgp graceful-перезапуск stalepath-time (1-4095)

Это команда, которая устанавливает максимальное время (в секундах) для удержания перезапуска устаревших путей однорангового узла.

Он также управляет расширенным таймером обновления маршрута.

Если эта команда настроена, и маршрутизатор не получает сообщение EoRR об обновлении маршрута, маршрутизатор удаляет устаревшие маршруты из таблицы BGP по истечении таймера. Таймер устаревшего пути запускается, когда маршрутизатор получает сообщение BoRR об обновлении маршрута.

bgp graceful-restart notification

Укажите поддержку плавного перезапуска для УВЕДОМЛЕНИЙ BGP.

После изменения этого параметра вам необходимо сбросить одноранговые узлы, чтобы объявить N-битную возможность плавного перезапуска.

Без возможности уведомления о повторном запуске (N-бит не установлен) GR не активируется при получении уведомлений об истечении времени прекращения / ОЖИДАНИЯ.

При отправке CEASE/Administrative Reset(clear bgp) сеанс закрывается, и маршруты не сохраняются. Когда установлен N-бит и bgp hard-administrative-resetотключен, активируется Graceful-Restart и маршруты сохраняются.

По умолчанию включено.

1.8.4.4.13.4 BGP Per Peer Graceful Restart

Возможность включать и отключать функции graceful restart, helper и no GR на всех режимах на одноранговом уровне.

Таким образом, bgp graceful restart может быть включен на глобальном уровне BGP или на уровне каждого узла. Существует два FSM, один для глобального режима BGP GR, а другой для одноранговой сети на GR.

Глобальный режим по умолчанию является вспомогательным, а одноранговый режим по умолчанию наследуется от глобального. Если настроен режим per peer, режим GR этого конкретного узла переопределит глобальный режим.

1.8.4.4.13.5 Команды глобального режима BGP GR

bgp graceful-restart

Эта команда включит функциональность BGP graceful restart на глобальном уровне.

bgp graceful-restart disable

Эта команда отключит как функциональность graceful restart, так и вспомогательный режим.

1.8.4.4.13.6 Команды однорангового режима BGP GR

neighbor A.B.C.D graceful-restart

Эта команда включит функцию плавного перезапуска BGP на одноранговом уровне.

neighbor A.B.C.D graceful-restart-helper

Эта команда включит функциональность BGP graceful restart helper только на одноранговом уровне.

neighbor A.B.C.D graceful-restart-disable

Эта команда отключит всю функциональность BGP graceful restart на одноранговом уровне.

1.8.4.4.13.7 Long-lived Graceful Restart

В настоящее время поддерживается только режим перезапуска. Эта возможность объявляется только в том случае, если согласована возможность плавного перезапуска.

bgp long-lived-graceful-restart stale-time (1-16777215)

Указывает максимальное время ожидания перед удалением устаревших маршрутов с длительным сроком службы для вспомогательных маршрутизаторов.

Значение по умолчанию равно 0, что означает, что функция по умолчанию отключена. Учитывается только плавный перезапуск.

1.8.4.4.13.8 Административное завершение работы

bgp shutdown[message MSG...]

Административное завершение работы всех одноранговых узлов экземпляра bgp. Удалите все одноранговые узлы BGP, но сохраните их конфигурации. Одноранговые узлы уведомляются в соответствии с RFC 8203 путем отправки NOTIFICATIONсообщения с кодом ошибки Ceaseи подкодом Administrative Shutdownперед завершением соединений. Это глобальное завершение работы не зависит от завершения работы соседнего узла, что означает, что его отмена не повлияет на отдельные отключенные одноранговые узлы.

Может быть указано необязательное сообщение о завершении работы.

1.8.4.4.14 Networks

network A.B.C.D/M

Эта команда добавляет сеть объявлений.

```
router bgp 1
 address-family ipv4 unicast
 network 10.0.0.0/8
 exit-address-family
```

В этом примере конфигурации говорится, что сеть 10.0.0.0 / 8 будет объявлена всем соседям. Маршрутизаторы некоторых производителей не рекламируют маршруты, если они отсутствуют в их таблицах маршрутизации IGP; *bgpd* не заботится о маршрутах IGP при объявлении своих маршрутов.

networkimport-check

Эта конфигурация изменяет поведение инструкции network. Если вы настроили это, базовая сеть должна существовать в rib. Если у вас настроена форма [нет], то BGP не будет проверять наличие сетей в rib. Для версий 7.3 и до того, как значения по умолчанию frr для центра обработки данных были такими, что сеть должна существовать, traditional не проверял наличие. Для версий 7.4 и выше, как для традиционных, так и для центров обработки данных, сеть должна существовать.

1.8.4.4.15 Поддержка IPv6

neighbor A.B.C.D activate

Эта конфигурация изменяет, следует ли включать семейство адресов для определенного соседа. По умолчанию включено только семейство одноадресных адресов IPv4.

```
router bgp 1
 address-family ipv6 unicast
 neighbor  2001:0DB8::1 activate
 network  2001:0DB8:5009::/64
 exit-address-family
```

В этом примере конфигурации говорится, что будет объявлено о сети 2001:0DB8:5009::/64 , и это позволяет соседу 2001:0DB8::1 получить это объявление.

По умолчанию всем соседям объявляется только семейство одноадресных адресов IPv4. Использование конфигурации "нет bgp по умолчанию ipv4-unicast" переопределяет это значение по умолчанию, так что все семейства адресов должны быть включены явно.

```
router bgp 1
 no bgp default ipv4-unicast
 neighbor  10.10.10.1 remote-as 2
 neighbor  2001:0DB8::1 remote-as 3
 address-family ipv4 unicast
 neighbor  10.10.10.1 activate
 network  192.168.1.0 /24
 exit-address-family
 address-family ipv6 unicast
 neighbor  2001:0DB8::1 activate
 network  2001:0DB8:5009::/64
 exit-address-family
```

Эта конфигурация демонстрирует, как "ipv4-unicast по умолчанию без bgp" может использоваться в настройке с двумя восходящими потоками, где каждый из восходящих потоков должен получать только объявления IPv4 или IPv6.

Используя bgp default ipv6-unicast конфигурация, семейство одноадресных адресов IPv6 включено по умолчанию для всех новых соседей.

1.8.4.4.16 Агрегирование маршрутов

1.8.4.4.16.1 Агрегирование маршрутов - семейство адресов IPv4

aggregate-адрес A.B.C.D / M

Эта команда задает совокупный адрес.

Для объявления агрегированного префикса в таблице BGP ДОЛЖЕН существовать более конкретный (более длинный) префикс. Например, если вы хотите создать aggregate-address 10.0.0.0/24вы должны убедиться, что у вас есть что-то вроде 10.0.0.5/32 или 10.0.0.0/26или любой другой префикс меньшего размера в таблице BGP. Таблицы информации о маршрутизации (RIB) недостаточно, вы должны перераспределить их в таблицу BGP.

aggregate-address A.B.C.D/M route-map

Примените карту маршрута для агрегированного префикса.

aggregate-address A.B.C.D/M origin<egp|igp|incomplete>

Переопределить ИСТОЧНИК для агрегированного префикса.

aggregate-address A.B.C.D/M as-set

Эта команда задает совокупный адрес. Результирующие маршруты включают в себя как установленные.

aggregate-address A.B.C.D/M summary-only

Эта команда задает совокупный адрес.

Более длинные префиксы реклама более конкретных маршрутов для всех соседей подавляется.

aggregate-address A.B.C.D/M matching-MED-only

Настройте агрегированный адрес так, чтобы он создавался только при совпадении средних маршрутов, в противном случае агрегированный маршрут не будет создан.

aggregate-address A.B.C.D/M suppress-map

Аналогично только краткое изложение, но будет подавлять только более конкретные маршруты, которые соответствуют выбранной карте маршрутов.

В этом примере конфигурации настраивается aggregate-address в соответствии с семейством адресов ipv4.

```
router bgp 1
 address-family ipv4 unicast aggregate-address  10.0.0.0/8
  aggregate-address  20.0.0.0/8 as-set  aggregate-address  40.0.0.0 /8 summary-only
aggregate-address  50.0.0.0 / 8 route-map aggr-rmap  exit-address-family
```

1.8.4.4.16.2 Агрегирование маршрутов - семейство адресов IPv6

aggregate-address X:X::X:X/M

Эта команда задает совокупный адрес.

aggregate-address X:X::X:X/M route-map

Примените карту маршрута для агрегированного префикса.

aggregate-address X:X::X:X/M origin <egp|igp|incomplete>

Переопределить ИСТОЧНИК для агрегированного префикса.

aggregate-address X:X::X:X/M as-set

Эта команда задает совокупный адрес. Результирующие маршруты включают в себя как установленные.

aggregate-address X:X::X:X/M summary-only

Эта команда задает совокупный адрес.

Более длинные префиксы реклама более конкретных маршрутов для всех соседей подавляется.

aggregate-address X:X::X:X/M matching-MED-only

Настройте агрегированный адрес так, чтобы он создавался только при совпадении средних маршрутов, в противном случае агрегированный маршрут не будет создан.

aggregate-address X:X::X:X/M suppress-map NAME

Аналогично только для сводки, но будет подавлять только более конкретные маршруты, которые соответствуют выбранной карте маршрутов.

В этом примере конфигурации устанавливается aggregate-addressсемейство адресов ipv6.

```
router bgp 1
 address-family ipv6 unicast
  aggregate-address 10::0/64
  aggregate-address 20::0/64 as-set
  aggregate-address 40::0/64 summary-only
  aggregate-address 50::0/64 route-map aggr-rmap
 exit-address-family
```

1.8.4.4.17 Перераспределение

Конфигурация перераспределения должна быть размещена в `address-family` разделе для конкретной AF для перераспределения. Доступность протокола для распространения определяется BGP AF; например, вы не можете распространять OSPFv3 в `address-family ipv4 unicast`, поскольку OSPFv3 поддерживает IPv6.

redistribute<babel|connected|eigrp|isis|kernel|openfabric|ospf|ospf6|rip|ripng|sharp|static|table>

[metric (0-4294967295)] [route-map WORD]

Перераспределение маршрутов из других протоколов в BGP.

redistribute vnc-direct

Перераспределите прямые (не через zebra) маршруты VNC в процесс BGP.

bgp

bgp update-delay MAX-DELAYESTABLISH-WAIT

Эта функция используется для включения режима только для чтения при перезапуске процесса BGP или при очистке процесса BGP с помощью ‘очистить ip bgp *’. Обратите внимание, что эта команда настроена на глобальном уровне и применяется ко всем экземплярам bgp / vrfs. Его нельзя использовать одновременно с командой “update-delay”, описанной ниже, которая вводится в каждом экземпляре bgp / vrf, который требуется для задержки установки обновлений и рекламы. Глобальный и индивидуальный подходы к определению задержки обновления являются взаимоисключающими.

Когда это применимо, режим только для чтения начнется, как только первый одноранговый узел достигнет установленного статуса и запустится таймер на секунды с максимальной задержкой. В этом режиме BGP не запускает какой-либо наилучший путь и не генерирует никаких обновлений для своих одноранговых узлов. Этот режим продолжается до:

1. Все настроенные одноранговые узлы, за исключением завершающих работу одноранговых узлов, отправили явный EOR (End-Of-RIB) или неявный EOR. Первое сохранение активности после того, как BGP достигнет установленного, считается

неявным-EOR. Если задано необязательное значение ожидания установления, то BGP будет ждать, пока одноранговые узлы достигнут установленного с начала задержки обновления до окончания периода ожидания установления, т.Е. Минимальный набор установленных одноранговых узлов, для которых ожидается EOR, будет одноранговыми узлами, установленными во время окна ожидания установления, а необязательно все настроенные соседи.

2. Период максимальной задержки закончился.

При выполнении любого из двух вышеуказанных условий BGP возобновляет процесс принятия решения и генерирует обновления для своих одноранговых узлов.

Максимальная задержка по умолчанию равна 0, т. Е. Функция по умолчанию отключена.

update-delay MAX-DELAY

update-delay MAX-DELAY ESTABLISH-WAIT

Эта функция используется для включения режима только для чтения при перезапуске процесса BGP или при очистке процесса BGP с помощью ‘очистить ip bgp *’. Обратите внимание, что эта команда настроена для конкретного экземпляра bgp / vrf, для которого включена эта функция. Его нельзя использовать одновременно с описанной выше глобальной “задержкой обновления bgp”, которая вводится на глобальном уровне и применяется ко всем экземплярам bgp. Глобальный и индивидуальный подходы к определению задержки обновления являются взаимоисключающими.

Когда это применимо, режим только для чтения начнется, как только первый одноранговый узел достигнет установленного статуса и запустится таймер на секунды с максимальной задержкой. В этом режиме BGP не запускает какой-либо наилучший путь и не генерирует никаких обновлений для своих одноранговых узлов. Этот режим продолжается до:

1. Все настроенные одноранговые узлы, за исключением завершающих работу одноранговых узлов, отправили явный EOR (End-Of-RIB) или неявный EOR. Первое сохранение активности после того, как BGP достигнет установленного, считается неявным-EOR. Если задано необязательное значение ожидания установления, то BGP будет ждать, пока одноранговые узлы достигнут установленного с начала задержки обновления до окончания периода ожидания установления, т.Е. Минимальный набор установленных одноранговых узлов, для которых ожидается EOR, будет одноранговыми узлами, установленными во время окна ожидания установления, а необязательно все настроенные соседи.
2. Период максимальной задержки закончился.

При выполнении любого из двух вышеуказанных условий BGP возобновляет процесс принятия решения и генерирует обновления для своих одноранговых узлов.

Максимальная задержка по умолчанию равна 0, т. Е. Функция по умолчанию отключена.

table-map ROUTE-MAP-NAME

Эта функция используется для применения карты маршрута при обновлении маршрута с BGP на Zebra. Разрешены все применимые операции сопоставления, такие как сопоставление по префиксу, следующему переходу, сообществам и т. Д. Операции установки для этой точки подключения ограничены только метрикой и следующим переходом. Любая операция этой функции не влияет на внутреннее ребро BGPs.

Поддерживается для семейств адресов ipv4 и ipv6. Он также работает с несколькими путями, однако настройка метрики основана только на наилучшем пути.

1.8.4.4.18 Одноранговые узлы

1.8.4.4.18.1 Определение одноранговых узлов

neighbor PEER remote-as ASN

Создает нового соседа, чей удаленный адрес является ASN. Одноранговый узел может быть адресом IPv4 или IPv6-адресом или интерфейсом, используемым для подключения.

```
router bgp 1
 neighbor  10.0.0.1 remote-as 2
```

В этом случае мой маршрутизатор в AS-1 пытается подключиться к AS-2 в 10.0.0.1.

Эта команда должна быть первой командой, используемой при настройке соседа. Если удаленный-как не указан, bgpd будет жаловаться следующим образом:

```
can't find neighbor 10.0.0.1
```

neighbor PEER remote-as internal

Создайте одноранговый узел, как если бы вы указывали ASN, за исключением того, что если ASN одноранговых узлов отличается от моего, как указано в `router bgp ASN` по команде соединение будет отклонено.

neighbor PEER remote-as external

Создайте одноранговый узел, как если бы вы указывали ASN, за исключением того, что если ASN одноранговых узлов совпадает с моим, как указано в `router bgp ASN` по команде соединение будет отклонено.

bgp listenrange<A.B.C.D/M|X:X::X:X/M> peer-group

Принимайте соединения от любых одноранговых узлов в указанном префиксе. Для настройки этих одноранговых узлов используется конфигурация из указанной одноранговой группы.

Примечание

При использовании диапазонов прослушивания BGP, если для связанной одноранговой группы настроена проверка подлинности TCP MD5, ваше ядро должно поддерживать это в префиксах. В Linux эта поддержка была добавлена в версии ядра 4.14. Если ваше ядро не поддерживает эту функцию, вы получите предупреждение в файле журнала, и диапазон прослушивания будет принимать соединения только от одноранговых узлов без настроенного MD5.

Кроме того, мы заметили, что при использовании этой опции в масштабе (несколько сотен одноранговых узлов) ядро может превысить свой лимит памяти. В этой ситуации вы увидите сообщения об ошибках, такие как:

bgpd: sockopt_tcp_signature: setsockopt(23): Cannot allocate memory

В этом случае вам необходимо увеличить значение sysctlnet.core.optmem_max, чтобы позволить ядру выделить необходимую память для опций.

bgp listenlimit<1-65535>

Определите максимальное количество одноранговых узлов, допустимых для одного экземпляра BGP. По умолчанию это ограничение равно 100. Увеличение этого значения действительно будет возможно, если в процессе BGP будет доступно больше файловых

дескрипторов. Это значение определяется базовой системой (значение ulimit) и может быть переопределено с помощью *–limit-fds* . Более подробная информация доступна в главе (Общие параметры вызова).

coalesce-time (0-4294967295)

Время в миллисекундах, которое BGP задержит, прежде чем решить, какие одноранговые узлы можно объединить в группу обновлений, чтобы сгенерировать для них одно обновление. Время по умолчанию равно 1000.

1.8.4.4.18.2 Настройка одноранговых узлов

neighbor PEER shutdown[message MSG...] [rtt (1-65535) [count (1-255)]]

Завершите работу одноранговый узла. Мы можем удалить конфигурацию соседano neighbor PEER remote-as ASN, но вся конфигурация соседа будет удалена. Если вы хотите сохранить конфигурацию, но хотите удалить узел BGP, используйте этот синтаксис.

При необходимости вы можете указать сообщение о завершении работы.

Кроме того, вы можете указать необязательно rttb миллисекундах, чтобы автоматически отключить одноранговый узел, если время обхода становится больше заданного.

Дополнительным countпараметром является количество сообщений keepalive для подсчета перед завершением работы одноранговый узла, если время обхода становится больше заданного.

neighbor PEER disable-connected-check

Разрешить пиринговые соединения между напрямую подключенными одноранговыми узлами eBGP с использованием петлевых адресов.

neighbor PEER disable-link-bw-encoding-ieee

По умолчанию пропускная способность в расширенных сообществах передается в формате IEEE с плавающей запятой, что соответствует проекту.

Более старые версии имеют реализацию, в которой значение расширенной полосы пропускания сообщества передается в кодировке uint32. Чтобы включить обратную совместимость, нам нужно отключить опцию кодирования IEEE с плавающей запятой для каждого узла.

neighbor PEER extended-optional-parameters

Принудительно используйте формат длины расширенных необязательных параметров для ОТКРЫТЫХ сообщений.

По умолчанию он отключен. Если длина стандартных необязательных параметров превышает один октет (255), то автоматически включается расширенный формат.

Для целей тестирования расширенный формат может быть включен с помощью этой команды.

neighbor PEERebgp-multihop

Указание ebgp-multihop позволяет устанавливать сеансы с соседними eBGP, когда они находятся на расстоянии нескольких переходов. Если сосед не подключен напрямую и этот регулятор не включен, сеанс не будет установлен.

Если IP-адрес одноранговый узла отсутствует в RIB и доступен по маршруту по умолчанию, тогда вы должны включить ip nht resolve-via-default.

neighbor PEER description...

Установите описание одноранговый узла.

neighbor PEER interfaceIFNAME

При подключении к одноранговому узлу BGP по локальному адресу IPv6 необходимо указать IFNAME интерфейса, используемого для подключения. Чтобы указать адреса сеансов IPv4, см. `neighbor PEER update-source` Команду ниже.

neighbor PEER interface remote-as <internal|external|ASN>

Настройте ненумерованный одноранговый узел BGP. `PEER` должно быть имя интерфейса. Сеанс будет установлен через локальные каналы связи IPv6. Используйте `internal` для сеансов iBGP и `external` eBGP или укажите ASN, если хотите.

neighbor PEER next-hop-self [force]

Эта команда определяет nexthop объявленного маршрута как эквивалентный адресу маршрутизатора bgp, если он изучен через eBGP. Это также позволит обойти сторонние переходы следующего поколения в пользу локального адреса bgp. Если указано необязательное ключевое `force` слово, модификация выполняется также для маршрутов, изученных через iBGP.

neighbor PEER attribute-unchanged [{as-path|next-hop|med}]

Эта команда указывает атрибуты, которые следует оставить неизменными для рекламных объявлений, отправляемых одноранговому узлу. Используйте это, чтобы оставить следующий переход без изменений в конфигурациях ipv6, поскольку директива route-map оставить следующий переход без изменений доступна только для ipv4.

neighbor PEER update-source <IFNAME|ADDRESS>

Укажите адрес источника IPv4, который будет использоваться для сеанса BGP с этим соседом, может быть указан либо как адрес IPv4 напрямую, либо как имя интерфейса (в этом случае демон *zebra* ДОЛЖЕН быть запущен, чтобы *bgpd* мог получить состояние интерфейса).

```
router bgp 64555
 neighbor foo update-source 192.168.0.1
 neighbor bar update-source lo0
```

neighbor PEER default-originate [route-map]

По умолчанию bgpd не объявляет маршрут по умолчанию (0.0.0.0 / 0), даже если он есть в таблице маршрутизации. Если вы хотите объявить одноранговому узлу маршруты по умолчанию, используйте эту команду.

Если route-маруказано ключевое слово, то маршрут по умолчанию будет создан только при соблюдении условий карты маршрута. Например, объявляйте маршрут по умолчанию, только если 10.10.10.10/32маршрут существует, и задайте произвольное сообщество для маршрута по умолчанию.

```
router bgp 64555
 address-family ipv4 unicast
  neighbor 192.168.255.1 default-originate route-map default
!
ip prefix-list p1 seq 5 permit 10.10.10.10/32
!
route-map default permit 10
 match ip address prefix-list p1
 set community 123:123
!
```

neighbor PEER port PORT

neighbor PEER password PASSWORD¶

Установите пароль MD5, который будет использоваться с сокетом tcp, который используется для подключения к удаленному узлу. Пожалуйста, обратите внимание, что если вы используете эту команду с большим количеством одноранговых узлов в Linux, вам следует рассмотреть возможность изменения системного идентификатора *net.core.*optmem_max на большее значение, чтобы избежать ошибок нехватки памяти в ядре Linux.

neighbor PEER send-community

neighbor PEER weight WEIGHT

Эта команда задает значение *веса* по умолчанию для маршрутов соседей.

neighbor PEER maximum-prefixNUMBER[force]

Задает максимальное количество префиксов, которые мы можем получить от данного одноранговой узла. Если это число будет превышено, сеанс BGP будет уничтожен.

На практике обычно предпочтительнее использовать список префиксов для ограничения того, какие префиксы принимаются от одноранговой узла, вместо использования этого регулятора. Прерывание сеанса BGP при превышении лимита гораздо более разрушительно, чем просто отклонение нежелательных префиксов. Метод списка префиксов также намного более детализирован и предлагает гораздо более разумный критерий соответствия, чем количество полученных префиксов, что делает его более подходящим для реализации политики.

Если forcesзадано, то ВСЕ префиксы учитываются как максимальные, а не только принятые. Это полезно для случаев, когда применяется фильтр входящих сообщений, но вы хотите, чтобы maximum-prefix действовал на ВСЕ (включая отфильтрованные) префиксы. Для этого параметра требуется, чтобы входящая программная реконфигурация была включена для одноранговой узла.

neighbor PEER maximum-prefix-outNUMBER¶

Задает максимальное количество префиксов, которые мы можем отправить данному одноранговому узлу.

Поскольку количество отправленных префиксов управляется группами обновлений, этот параметр создает отдельную группу обновлений для исходящих обновлений.

neighbor PEER local-as AS-NUMBER [no-prepend] [replace-as]

Укажите альтернативный AS для этого процесса BGP при взаимодействии с указанным одноранговым узлом. Без модификаторов указанный local-as добавляется к полученному AS_PATH при получении обновлений маршрутизации от одноранговой узла и добавляется к исходящему AS_PATH (после процесса local AS) при передаче локальных маршрутов одноранговому узлу.

Если указан атрибут no-prepend, то предоставленный local-as не добавляется к полученному AS_PATH.

Если указан атрибут replace-as, то при передаче обновлений локального маршрута этому узлу добавляется только предоставленный локальный AS_PATH к AS_PATH.

Обратите внимание, что replace-as может быть указан только в том случае, если нет-prepend .

Эта команда разрешена только для одноранговых узлов eBGP.

neighbor <A.B.C.D|X:X::X:X|WORD> as-override

Переопределите номер AS исходного маршрутизатора локальным номером AS.

Обычно эта конфигурация используется в PEs (граница поставщика) для замены входящего клиента в качестве номера, чтобы подключенный CE (граница клиента) мог использовать тот же номер AS, что и другие сайты клиентов. Это позволяет клиентам сети провайдера использовать один и тот же номер AS на своих сайтах.

Эта команда разрешена только для одноранговых узлов eBGP.

neighbor <A.B.C.D|X:X::X:X|WORD> allowas-in [<(1-10)|origin>]

Принимайте входящие маршруты с помощью пути AS, содержащего номер AS с тем же значением, что и текущая система AS.

Это используется, когда вы хотите использовать тот же номер, что и на ваших сайтах, но вы не можете подключить их напрямую. Это альтернатива соседнему СЛОВУ как переопределению.

Параметр (1-10) настраивает количество принятых вхождений системы КАК число в пути AS.

Параметр origin настраивает BGP на прием только маршрутов, созданных с тем же номером AS, что и в системе.

Эта команда разрешена только для одноранговых узлов eBGP.

neighbor <A.B.C.D|X:X::X:X|WORD> addpath-tx-all-paths

Настройте BGP для отправки всех известных путей соседним, чтобы сохранить возможности нескольких путей внутри сети.

neighbor <A.B.C.D|X:X::X:X|WORD> addpath-tx-bestpath-per-AS

Настройте BGP для отправки наиболее известных путей соседним, чтобы сохранить возможности нескольких путей внутри сети.

neighbor <A.B.C.D|X:X::X:X|WORD> disable-addpath-rx

Не принимайте дополнительные пути от этого соседа.

neighbor PEER ttl-security hopsNUMBER

Эта команда применяет обобщенный механизм безопасности TTL (GTSM), как указано в RFC 5082. С помощью этой команды только соседям, которые находятся на расстоянии указанного количества переходов, будет разрешено стать соседями. Эта команда является взаимоисключающей с *ebgp-multihop*.

neighbor PEER capabilityextended-nexthop

Разрешить bgp согласовывать расширенные возможности nexthop с его одноранговым узлом. Если вы просматриваете LL-адрес версии 6, эта возможность включается автоматически. Если вы просматриваете глобальный адрес версии 6, то включение этой команды позволит BGP устанавливать маршруты версии 4 с помощью v6 nexthops, если у вас не настроен v4 на интерфейсах.

neighbor <A.B.C.D|X:X::X:X|WORD> accept-own

Включить обработку самостоятельно созданных VPN-маршрутов, содержащих accept-own сообщество.

Эта функция позволяет обрабатывать самостоятельно созданные VPN-маршруты, которые динамик BGP получает от отражателя маршрута. Маршрут, созданный самим пользователем, - это маршрут, который изначально был объявлен самим пользователем. Согласно RFC 4271, динамик BGP отклоняет рекламные объявления, созданные самим динамиком. Однако механизм BGP ACCEPT_OWN позволяет маршрутизатору принимать объявленные им префиксы при отражении от отражателя маршрута, который изменяет определенные атрибуты префикса.

Специальное вызываемое сообщество accept-ownприсоединяется к префиксу с помощью отражателя маршрута, который является сигналом для принимающего маршрутизатора для обхода проверки ORIGINATOR_ID и NEXTHOP / MP_REACH_NLRI.

По умолчанию: отключен.

neighbor <A.B.C.D|X:X::X:X|WORD> path-attribute discard(1-255)...

Удаляет указанные атрибуты пути из сообщений ОБНОВЛЕНИЯ BGP от указанного соседа.

Если вам не нужны определенные атрибуты, вы можете удалить их с помощью этой команды и позволить BGP продолжить, игнорируя эти атрибуты.

neighbor <A.B.C.D|X:X::X:X|WORD> graceful-shutdown

Пометьте все маршруты от этого соседа как менее предпочтительные, установив graceful-shutdownзначение community и local-preference равным 0.

bgp fast-external-failover

Эта команда заставляет bgp немедленно отключать одноранговые узлы ebgp при разрыве соединения. по умолчанию используется *bgp fast-external-failover*, который не будет отображаться как часть *демонстрационного запуска*. Форма команды "нет" отключает эту возможность.

bgp defaultipv4-unicast

Эта команда позволяет пользователю указать, включено семейство одноадресных адресов IPv4 по умолчанию или нет. По умолчанию эта команда включена и не отображается. Отображается форма команды *no bgp default ipv4-unicast*.

bgp default ipv4-multicast

Эта команда позволяет пользователю указать, включено ли семейство многоадресных адресов IPv4 по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp для многоадресной рассылки ipv4 по умолчанию*.

bgp defaultipv4-vpn

Эта команда позволяет пользователю указать, включено ли семейство адресов VPN IPv4 MPLS по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp ipv4-vpn по умолчанию*.

bgp defaultipv4-flowspec

Эта команда позволяет пользователю указать, включено ли семейство адресов IPv4 Flowspec по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp ipv4-flowspec по умолчанию*.

bgp default ipv6-unicast

Эта команда позволяет пользователю указать, включено ли семейство одноадресных адресов IPv6 по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp для одноадресной передачи ipv6 по умолчанию*.

bgp default ipv6-multicast

Эта команда позволяет пользователю указать, включено ли по умолчанию семейство многоадресных адресов IPv6 или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp для групповой рассылки ipv6 по умолчанию*.

bgp default ipv6-vpn

Эта команда позволяет пользователю указать, включено ли семейство адресов VPN IPv6 MPLS по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp ipv6-vpn по умолчанию*.

bgp default ipv6-flowspec

Эта команда позволяет пользователю указать, включено ли семейство адресов IPv6 Flowspec по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp ipv6-flowspec по умолчанию*.

bgp default l2vpn-evpn

Эта команда позволяет пользователю указать, включено семейство адресов L2VPN EVPN по умолчанию или нет. По умолчанию эта команда отключена и не отображается. Отображается форма команды *bgp l2vpn-evpn по умолчанию*.

bgp default show-hostname

Эта команда показывает имя узла однорангового узла в выводах определенных команд BGP. Устранение неполадок проще, если у вас есть несколько одноранговых узлов BGP.

bgp default show-nexthop-hostname

Эта команда показывает имя хоста следующего перехода в выводах некоторых команд BGP. Устранение неполадок проще, если у вас есть несколько одноранговых узлов BGP и несколько маршрутов для проверки.

neighbor PEER advertisement-interval (0-600)

Настройте минимальный интервал объявления маршрута (mrai) для рассматриваемого однорангового узла. Это число составляет от 0 до 600 секунд, при этом интервал объявления по умолчанию равен 0.

neighbor PEER timers (0-65535) (0-65535)

Установите таймеры сохранения и удержания для соседа. Первое значение - keepalive, а второе - время удержания.

neighbor PEER timers connect(1-65535)

Установите таймер подключения для соседа. Таймер подключения определяет, сколько времени BGP ожидает между попытками подключения к соседнему серверу.

neighbor PEER timersdelayopen(1-240)

Эта команда позволяет пользователю включить *RFC 4271 <https://tools.ietf.org/html/rfc4271 />* DelayOpenTimer с указанным интервалом или отключите его с помощью команды отрицания для однорангового узла. По умолчанию параметр DelayOpenTimer отключен. Интервал таймера может быть установлен на длительность от 1 до 240 секунд.

bgp minimum-holdtime(1-65535)

Эта команда позволяет пользователю запретить установление сеанса с одноранговыми узлами BGP с меньшим временем ожидания, меньшим, чем настроенное минимальное время ожидания. Когда эта команда не задана, минимальное время ожидания не работает.

bgp tcp-keepalive (1-65535) (1-65535) (1-30)

Эта команда позволяет пользователю настроить TCP keepalive с новыми одноранговыми узлами BGP. Каждый параметр соответственно обозначает TCP keepalive таймер ожидания (секунды), интервал (секунды) и максимальное количество проб. По умолчанию TCP keepalive отключен.

1.8.4.4.18.3 Отображение информации об одноранговых узлах

show bgp <afi> <safi> neighbors WORD bestpath-routes [detail] [json] [wide]

Для данного соседа, WORD, который указан, перечислены маршруты, выбранные BGP как имеющие наилучший путь.

Если detailопция указана, будет отображена подробная версия всех маршрутов. **show [ip] bgp [afi] [safi] PREFIX** будет использоваться тот же формат, что и для всей таблицы полученных, объявленных или отфильтрованных префиксов.

Если json опция указана, выходные данные отображаются в формате JSON.

Если wide опция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.

1.8.4.4.18.4 Одноранговая фильтрация

neighbor PEER distribute-listNAME[in|out]

Эта команда определяет список рассылки для одноранговой узла. *прямой* - это `in` или `out`.

neighbor PEER prefix-list NAME[in|out]

neighbor PEER filter-list NAME[in|out]

neighbor PEER route-map NAME[in|out]

Примените карту маршрута к соседнему объекту. *direct* должен быть включен или *отключен*.

bgp route-reflector allow-outbound-policy

По умолчанию изменение атрибута с помощью политики вывода карты маршрутов не отражается на отраженных маршрутах. Этот параметр также позволяет отразить изменения. После включения она влияет на все отраженные маршруты.

neighbor PEER sender-as-path-loop-detection

Включите обнаружение стороны отправителя в виде циклов путей и отфильтруйте неверные маршруты перед их отправкой.

По умолчанию этот параметр отключен.

1.8.4.4.18.5 Одноранговые группы

Одноранговые группы используются для улучшения масштабирования путем создания одинаковой обновленной информации для всех членов одноранговой группы. Обратите внимание, что это означает, что маршруты, сгенерированные членом одноранговой группы, будут отправлены обратно этому исходному одноранговому узлу с атрибутом идентификатора отправителя, установленным для указания исходного одноранговой узла. Все одноранговые узлы, не связанные с определенной группой одноранговых узлов, рассматриваются как принадлежащие к группе одноранговых узлов по умолчанию и будут обмениваться обновлениями.

neighbor WORDpeer-group

Эта команда определяет новую одноранговую группу.

neighbor PEER peer-group PGNAME

Эта команда привязывает определенное слово одноранговой группы к одноранговой.

neighbor PEER solo

Эта команда используется для указания на то, что маршруты, объявленные одноранговым узлом, не должны отражаться обратно одноранговому узлу. Эта команда имеет смысл только тогда, когда в одноранговой группе определен один одноранговый узел.

show [ip] bgp peer-group[json]

Эта команда отображает настроенные одноранговые группы BGP.

```
exit1-debian-9# show bgp peer-group

BGP peer-group test1, remote AS 65001
Peer-group type is external
```

```
Configured address-families: IPv4 Unicast; IPv6 Unicast;
1 IPv4 listen range(s)
 192.168.100.0/24
2 IPv6 listen range(s)
 2001:db8:1::/64
   2001:db8:2::/64
Peer-group:
 192.168.200.1.168.200.1 Active
 2001:db8::1 Active

BGP peer-group test2
Peer-group type is external
Configured address-families: IPv4 Unicast;
```

Необязательный `json` параметр используется для отображения выходных данных JSON.

```
{ "test1":{
 "remoteAs": 65001, "type":"external",
 "addressFamiliesConfigured": [
 "IPv4 Unicast", "IPv6 Unicast"
], "dynamicRanges":{
 "IPv4": {
 "count": 1, "ranges": [
"192.168.100.0 \ / 24"
 ]
 }, "IPv6": {
 "count": 2, "ranges": [
 "2001:db8:1::\/64", "2001:db8:2::\/64"
 ]
 }
 }, "members":{
 "192.168.200.1":{
 "status":"Active"
 }, "2001:db8::1":{
 "status":"Active"
 }
 }
 }, "test2":{
 "type":"external",
 "addressFamiliesConfigured":[
 "IPv4 Unicast"
 ]
 }
}
```

1.8.4.4.18.6 Согласование возможностей

neighbor PEER strict-capability-match

Строго сравнивает удаленные и локальные возможности. Если возможности отличаются, отправьте сообщение об ошибке неподдерживаемой возможности, а затем сбросьте соединение.

Возможно, вы захотите отключить отправку необязательного параметра ОТКРЫТОГО сообщения о согласовании возможностей одноранговому узлу, когда удаленный одноранговый

узел не реализует согласование возможностей. Пожалуйста, используйте команду dont-capability-negotiate, чтобы отключить эту функцию.

neighbor PEER dont-capability-negotiate

Подавить согласование возможности отправки в качестве необязательного параметра ОТКРЫТОГО сообщения одноранговому узлу. Эта команда влияет только на одноранговую конфигурацию, отличную от конфигурации одноадресной рассылки IPv4.

Когда удаленный одноранговый узел не имеет функции согласования возможностей, удаленный одноранговый узел вообще не будет отправлять какие-либо возможности. В этом случае bgp настраивает одноранговый узел с настроенными возможностями.

Вы можете предпочесть локально настроенные возможности больше, чем согласованные возможности, даже несмотря на то, что удаленный узел отправляет возможности. Если одноранговый узел настроен с помощью переопределения возможностей, bgpd игнорирует полученные возможности, а затем переопределяет согласованные возможности с помощью настроенных значений.

Кроме того, оператору следует напомнить, что эта функция принципиально отключает возможность использования широко распространенных функций BGP. BGP без номера, поддержка имен хостов, AS4, Addpath, обновление маршрута, ORF, динамические возможности и плавный перезапуск.

neighbor PEER override-capability

Переопределите результат согласования возможностей с помощью локальной конфигурации. Игнорируйте значение возможностей удаленного узла.

1.8.4.4.19 AS Path Access Lists

ПОСКОЛЬКУ список доступа к пути определяется пользователем КАК путь.
bgp as-path access-list WORD [seq (0-4294967295)] permit|deny LINE

Эта команда определяет новый список доступа к пути AS.
show bgp as-path-access-list [json]

Отображать все BGP В ВИДЕ списков доступа к путям.
Если json опция указана, выходные данные отображаются в формате JSON.
show bgp as-path-access-list WORD[json]

Отобразить указанный BGP В ВИДЕ списка доступа к пути.
Если json опция указана, выходные данные отображаются в формате JSON.

1.8.4.4.19.1 Пример конфигурации политики фильтрации Bogon ASN

```
6  bgp as-path access-list 99 permit _0_
7  bgp as-path access-list 99 permit _23456_
8  bgp as-path access-list 99 permit _1310[0-6][0-9]_|_13107[0-1]_
9  bgp as-path access-list 99 seq 20 permit ^65
```

1.8.4.4.20 Использование в качестве пути в карте маршрута

match as-path

Для заданного as-path, WORD, сопоставьте его с BGP as-path, указанным для префикса, и, если он совпадает, выполните обычные действия с картой маршрута. Форма no команды удаляет это соответствие с карты маршрута.

set as-path prepend AS-PATH

Добавьте заданную строку чисел AS к AS_PATH NLRI пути BGP. Форма no этой команды удаляет эту операцию set из карты маршрута.

set as-path prepend last-as NUM

Добавьте заданную строку чисел AS к AS_PATH NLRI пути BGP. Форма no этой команды удаляет эту операцию set из карты маршрута.

set as-path replace <any|ASN>

Замените определенный номер AS на локальный номер AS. any заменяет каждый номер AS в AS-PATH на локальный номер AS.

1.8.4.4.21 Атрибут сообществ

Атрибут сообществ BGP широко используется для реализации маршрутизации политики. Сетевые операторы могут манипулировать атрибутом сообществ BGP на основе своей сетевой политики. Атрибут сообществ BGP определяется в RFC 1997 и RFC 1998. Это необязательный транзитивный атрибут, поэтому локальная политика может проходить через другую автономную систему.

Атрибут сообществ представляет собой набор значений сообществ. Каждое значение сообщества имеет длину 4 октета. Для определения значения сообщества используется следующий формат.

AS:VAL

Этот формат представляет значение сообществ 4 октета. `AS` представляет собой октет высокого порядка 2 в цифровом формате. `VAL` представляет собой октет младшего порядка 2 в цифровом формате. Этот формат полезен для определения КАК ориентированного значения политики. Например, `7675:80` может использоваться, когда AS 7675 хочет передать значение локальной политики 80 соседнему узлу.

Internet

`internet` представляет значение 0 для известных сообществ.

graceful-shutdown

`graceful-shutdown` представляет известную ценность для сообществ `GRACEFUL_SHUTDOWN` `0xFFFF0000` `65535:0`. В **RFC 8326** реализована цель Graceful BGP Session Shutdown для уменьшения количества потерянного трафика при отключении сеансов BGP для обслуживания. Использование сообщества должно поддерживаться со стороны ваших коллег, чтобы действительно иметь какой-либо эффект.

accept-own

`accept-own` представляет известную ценность для сообществ `ACCEPT_OWN` `0xFFFF0001` `65535:1`. В **RFC 7611** реализован способ передачи маршрутизатору сигнала о принятии маршрутов с локальным адресом nexthop. Это может иметь место при выполнении политики и наличии трафика, имеющего nexthop, расположенного в другом VRF, но все еще локального интерфейса к маршрутизатору. Рекомендуется прочитать RFC для получения полной информации.

route-filter-translated-v4

`route-filter-translated-v4` представляет известную ценность для сообществ `ROUTE_FILTER_TRANSLATED_v4` `0xFFFF0002` `65535:2`.

route-filter-v4

`route-filter-v4` представляет известную ценность для сообществ `ROUTE_FILTER_v4` `0xFFFF0003` `65535:3`.

route-filter-translated-v6

`route-filter-translated-v6` представляет известную ценность для сообществ `ROUTE_FILTER_TRANSLATED_v6` `0xFFFF0004` `65535:4`.

route-filter-v6

`route-filter-v6` представляет известную ценность для сообществ `ROUTE_FILTER_v6` `0xFFFF0005` `65535:5`.

llgr-stale

`llgr-stale` представляет известную ценность для сообществ `LLGR_STALE` `0xFFFF0006` `65535:6`. Назначен и предназначен только для использования с маршрутизаторами, поддерживающими возможность долговременного плавного перезапуска, как описано в [Черновик-IETF-uttaro-idr-bgp-persistence]. Маршрутизаторы, получающие маршруты с этим сообществом, могут (в зависимости от реализации) выбрать разрешить отклонять или изменять маршруты при наличии или отсутствии этого сообщества.

no-llgr

`no-llgr` представляет известную ценность для сообществ `NO_LLGR` `0xFFFF0007` `65535:7`. Назначен и предназначен только для использования с маршрутизаторами, поддерживающими возможность долговременного плавного перезапуска, как описано в [Черновик-IETF-uttaro-idr-bgp-persistence]. Маршрутизаторы, получающие маршруты с этим сообществом, могут (в зависимости от реализации) выбрать разрешить отклонять или изменять маршруты при наличии или отсутствии этого сообщества.

accept-own-nexthop

`accept-own-nexthop` представляет известную ценность для сообществ `accept-own-nexthop` `0xFFFF0008` `65535:8`. В [Draft-IETF-agreewal-idr-accept-own-nexthop] описывается, как помечать и маркировать VPN-маршруты, чтобы иметь возможность отправлять трафик между VRF через внутренний домен уровня 2 на одном и том же PE-устройстве. Для получения полной информации обратитесь к [Draft-IETF-agreewal-idr-accept-own-nexthop].

blackhole

`blackhole` представляет известную ценность для сообществ `BLACKHOLE` `0xFFFF029A` `65535:666`. **RFC 7999** документирует отправку префиксов одноранговым узлам EBGP и восходящему потоку с целью блокировки трафика. Префиксы, помеченные этим сообществом, обычно не должны повторно

рекламироваться соседями исходной сети. При получении `BLACKHOLE` сообщества от спикера BGP `NO_ADVERTISE` сообщество добавляется автоматически.

no-export

`no-export` представляет известную ценность для сообществ `NO_EXPORT` `0xFFFFFF01`. Все маршруты, содержащие это значение, не должны рекламироваться за пределами границ конфедерации BGP. Если соседний узел BGP является частью конфедерации BGP, узел рассматривается как находящийся внутри границы конфедерации BGP, поэтому маршрут будет объявлен узлу.

no-advertise

`no-advertise` представляет известную ценность для сообществ `NO_ADVERTISE` `0xFFFFFF02`. Все маршруты, содержащие это значение, не должны рекламироваться другим одноранговым узлам BGP.

local-AS

`local-AS` представляет известную ценность для сообществ `NO_EXPORT_SUBCONFED` `0xFFFFFF03`. Все маршруты, содержащие это значение, не должны объявляться внешним одноранговым узлам BGP. Даже если соседний маршрутизатор является частью конфедерации, он считается внешним узлом BGP, поэтому маршрут не будет объявлен узлу.

no-peer

`no-peer` представляет известную ценность для сообществ `NOPEER` `0xFFFFFF04` `65535:65284`. **RFC 3765** используется для передачи в другую сеть информации о том, как исходная сеть хочет, чтобы префикс распространялся.

При получении атрибута сообщества повторяющиеся значения сообщества в атрибуте игнорируются, а значение сортируется в числовом порядке.

Draft-IETF-uttaro-idr-bgp-persistence (1 ,2)

<https://tools.ietf.org/id/draft-uttaro-idr-bgp-persistence-04.txt >

Draft-IETF-agrewal-idr-accept-own-nexthop (1 ,2)

<https://tools.ietf.org/id/draft-agrewal-idr-accept-own-nexthop-00.txt >

1.8.4.4.21.1 Списки сообщества

Списки сообществ - это определяемые пользователем списки значений атрибутов сообщества. Эти списки можно использовать для сопоставления или изменения атрибута сообществ в сообщениях ОБ ОБНОВЛЕНИИ.

Существует два типа списков сообществ:

Standard

Этот тип принимает явное значение для атрибута.

Expanded

Этот тип принимает регулярное выражение. Поскольку регулярное выражение должно интерпретироваться при каждом использовании, расширенные списки сообщества работают медленнее, чем стандартные списки.

bgp community-list standard NAME permit|deny COMMUNITY

Эта команда определяет новый стандартный список сообщества. `COMMUNITY` является ли ценность сообщества. `COMMUNITY` Скомпилирована в структуру сообщества. Мы можем определить список нескольких сообществ под одним и тем же именем. В этом случае совпадение произойдет в порядке, определенном пользователем. Как только список сообществ совпадает с атрибутом сообществ в обновлениях BGP, он возвращает разрешение или запрет в соответствии с определением списка сообществ. Если нет совпадающей записи, будет возвращено значение deny. Когда `COMMUNITY` он пуст, он соответствует любым маршрутам.

bgp community-list expanded NAME permit|deny COMMUNITY

Эта команда определяет новый расширенный список сообщества. COMMUNITYявляется строковым выражением атрибута сообществ. COMMUNITYможет быть регулярным выражением (регулярные выражения BGP), соответствующим атрибуту сообщества в обновлениях BGP. Расширенное сообщество используется только для фильтрации, а не для задания действий.

Устарела с версии 5.0: рекомендуется использовать более явные версии этой команды.

bgp community-list NAME permit|deny COMMUNITY

Если тип списка сообщества не указан, тип списка сообщества определяется автоматически. Если COMMUNITYможет быть скомпилирован в атрибут сообщества, список сообщества определяется как стандартный список сообщества. В противном случае он определяется как расширенный список сообщества. Эта функция оставлена для обеспечения обратной совместимости. Использование этой функции не рекомендуется.

Обратите внимание, что все списки сообщества используют одно и то же пространство имен, поэтому нет необходимости указывать standardили expanded; эти модификаторы носят чисто эстетический характер.

show bgp community-list[NAME detail]

Отображает информацию о списке сообщества. Когда `NAME` указано, отображается информация из указанного списка сообществ.

```
# show bgp community-list
Named Community
standardlistCLISTpermit 7675:80  7675:100 no-export
 denyinternet
  Named Community expanded
listEXPANDpermit :

  # show bgp community-list CLIST detail
   NamedCommunity
standardlistCLISTpermit 7675:80  7675:100 no-export
 denyinternet
```

1.8.4.4.21.2 Пронумерованные списки сообществ

Когда номер используется для имени списка сообщества BGP, номер имеет особое значение. Номер списка сообщества в диапазоне от 1 до 99 является стандартным списком сообщества. Список сообществ в диапазоне от 100 до 500 - это расширенный список сообществ. Эти списки сообществ называются нумерованными списками сообществ. С другой стороны, обычные списки сообществ вызываются как именованные списки сообществ.

bgp community-list(1-99) permit|deny COMMUNITY

Эта команда определяет новый список сообществ. Аргумент (1-99) определяет идентификатор списка.

bgp community-list(100-500) permit|deny COMMUNITY

Эта команда определяет новый расширенный список сообщества. Аргумент (100-500) определяет идентификатор списка.

1.8.4.4.21.3 Псевдоним сообщества

Псевдонимы сообщества BGP полезны для быстрого определения, какие сообщества установлены для определенного префикса в удобочитаемом формате. Особенно удобно для огромного количества сообществ. Точно определенные псевдонимы могут помочь вам быстрее определять объекты на проводе.

bgp communityaliasNAMEALIAS

Эта команда создает псевдоним для сообщества, который будет использоваться позже в различных выводах командной строки в удобочитаемом формате.

```
~# vtysh -c 'show run' | grep 'bgp community alias'
bgp community alias 65001:14 community-1
bgp65001:123:1 lcommunity-1

~# vtysh -c 'show ip bgp 172.16.16.1/32'
BGP routing table entry for 172.16.16.1 / 32, version 21
Paths: (2 available, best #2, table default)
 Advertised to non peer-group peers:
 65030
    192.168.0.2.168.0.2 from 192.168.0.2 (172.16.16.1)
 Origin incomplete, metric 0, valid, external, best (Neighbor IP)
 Community: 65001:12 65001:13 community-1 65001:65534
 Large Community: lcommunity-1 65001:123:2
     Last update: Fri Apr 16  12:51:27  2021
```

show bgp [afi] [safi] [all] alias WORD [wide|json]

Отображение префиксов с соответствующим псевдонимом сообщества BGP.

1.8.4.4.21.4 Использование сообществ в маршрутных картах

В карты маршрутов мы можем сопоставить или установить атрибут сообществ BGP. Используя эту функцию, оператор сети может реализовать свою сетевую политику на основе атрибута сообществ BGP.

В картах маршрутов можно использовать следующие команды:

match alias WORD

Эта команда выполняет сопоставление с обновлениями BGP, используя псевдоним сообщества WORD. Когда значение одного из сообществ BGP совпадает со значением псевдонима сообщества в псевдониме сообщества, оно совпадает.

match community WORDexact-match[exact-match]

Эта команда выполняет сопоставление с обновлениями BGP, используя псевдоним сообщества WORD. Когда значение одного из сообществ BGP совпадает со значением псевдонима сообщества в псевдониме сообщества, оно совпадает.

set community <none|COMMUNITY>

Эта команда выполняет сопоставление с обновлениями BGP, используя слово списка сообщества. Когда значение одного из сообществ BGP совпадает со значением одного из сообществ в списке сообществ, оно совпадает. Когда *точное соответствие* ключевое слово указано, совпадение происходит только тогда, когда обновления BGP имеют полностью то же значение сообщества, указанное в списке сообщества.

Если none указывается как значение сообщества, атрибут сообщества не отправляется.

Невозможно установить расширенный список сообществ.

set comm-list

Эта команда удаляет значение сообществ из атрибута сообществ BGP. Это `word` имя списка сообщества. Когда значение сообществ маршрута BGP совпадает со списком сообществ, `word` значение сообществ удаляется. Когда все значения сообществ в конечном итоге удаляются, атрибут сообществ обновления BGP полностью удаляется.

1.8.4.4.21.5 Пример конфигурации

Следующая конфигурация является примером наиболее типичного использования атрибута сообществ BGP. В примере AS 7675 обеспечивает восходящее интернет-соединение с AS 100. Если в AS 7675 существует следующая конфигурация, оператор сети AS 100 может установить локальные предпочтения в сети AS 7675, установив атрибут сообществ BGP для обновлений.

```
router bgp 7675
 neighbor  192.168.0.1 remote-as 100
 address-family ipv4 unicast
 neighbor  192.168.0.1 route-map RMAP in
 exit-address-family
!
bgp community-list 70 permit 7675:70
bgp community-list 80 7675:80
bgp community-list 90 permit 7675:90
!
route-map 10
 match community 70
 set local-preference 70
!
route-map20
 match community 80
 set local-preference 80
!
route-map30
 match community 90
 set local-preference 90
```

Следующая конфигурация объявляет `10.0.0.0/8` от AS 100 до AS 7675. Маршрут имеет значение сообщества `7675:80`, поэтому, когда вышеуказанная конфигурация существует в AS 7675, значение локального предпочтения объявленных маршрутов будет равно 80.

```
router bgp 100
 network 10.0.0.0/8
```

```
 neighbor 192.168.0.2 remote-as 7675
 address-family ipv4 unicast
  neighbor 192.168.0.2 route-map RMAP out
 exit-address-family
!
ip prefix-list PLIST permit 10.0.0.0/8
!
route-map RMAP permit 10
 match ip address prefix-list PLIST
 set community 7675:80
```

Следующая конфигурация является примером фильтрации маршрутов BGP с использованием атрибута сообществ. Эта конфигурация разрешает только маршруты BGP, которые имеют значение BGP communities (`0:80` и `0:90`) или `0:100`. Оператор сети может установить специальное значение для внутренних сообществ на пограничном маршрутизаторе BGP, а затем ограничить объявления маршрута BGP во внутренней сети.

```
router bgp 7675
 neighbor  192.168.0.1 remote-as 100
 address-family ipv4 unicast
 neighbor  192.168.0.1 route-map RMAP in
 exit-address-family
!
bgp community-list 1 permit 0:80 0:90
bgp community-list 1 0: 100
!
route-map RMAP permit in
 match community 1
```

В следующем примере выполняется фильтрация маршрутов BGP, значение сообщества которых равно `1:1`. При отсутствии соответствия возвращается список сообщества `deny`. Чтобы избежать фильтрации всех маршрутов, а `permit` строка устанавливается в конце списка сообществ.

```
router bgp 7675
 neighbor  192.168.0.1 remote-as 100
 address-family ipv4 unicast
 neighbor  192.168.0.1 route-map RMAP in
 exit-address-family
!
bgp community-list standard FILTER deny 1:1
bgp community-list standard FILTER permit
!
route-map RMAP permit 10
 match community FILTER
```

Ключевое слово communities value internet имеет особые значения в стандартных списках сообщества. В приведенном ниже примере internet соответствует всем маршрутам BGP, даже если у маршрута вообще нет атрибута communities. Итак, список сообщества INTERNET является таким же, как FILTER в предыдущем примере.

```
bgp community-list standard INTERNET deny 1:1
bgp community-list standard INTERNET permit
internet
```

Следующая конфигурация является примером удаления значения сообществ. При такой конфигурации значения сообщества 100:1 и 100:2 удаляются из обновлений BGP. Для удаления

значений сообществ используется только permit список сообществ. deny список сообщества игнорируется.

```
router bgp 7675
 neighbor 192.168.0.1 remote-as 100
 address-family ipv4 unicast
  neighbor 192.168.0.1 route-map RMAP in
 exit-address-family
!
bgp community-list standard DEL permit 100:1
100:2
!
route-map RMAP permit 10
 set comm-list DEL delete
```

1.8.4.4.21.6 Атрибут расширенных сообществ

Атрибут расширенных сообществ BGP введен с технологией MPLS VPN / BGP. MPLS VPN / BGP расширяет возможности сетевой инфраструктуры для обеспечения функциональности VPN. В то же время для маршрутизации политик требуется новая структура. С атрибутом расширенных сообществ BGP мы можем использовать целевой маршрут или исходный сайт для реализации сетевой политики для MPLS VPN / BGP.

Атрибут расширенных сообществ BGP аналогичен атрибуту сообществ BGP. Это необязательный транзитивный атрибут. Атрибут расширенных сообществ BGP может содержать несколько значений расширенного сообщества. Каждое расширенное значение сообщества имеет длину в восемь октетов.

Атрибут расширенных сообществ BGP предоставляет расширенный диапазон по сравнению с атрибутом сообществ BGP. Добавление к этому поля типа в каждом значении обеспечивает структуру пространства сообщества.

Существует два формата для определения ценности расширенного сообщества. Один из них основан на формате AS, а другой - на формате, основанном на IP-адресе.

AS:VAL

Это формат, который определяется КАК основанный на расширенной ценности сообщества. ASчасть - это подполе глобального администратора на 2 октета в расширенном значении сообщества. VALчасть представляет собой подполе локального администратора на 4 октета. 7675:100представляет КАК 7675 значение политики 100.

IP-Address:VAL

Это формат для определения значения расширенного сообщества на основе IP-адреса. IP-Addressчасть представляет собой подполе глобального администратора на 4 октета. VALчасть представляет собой подполе локального администратора на 2 октета.

1.8.4.4.21.7 Расширенные списки сообщества

bgp extcommunity-list standard NAME permit|deny EXTCOMMUNITY

Эта команда определяет новый стандартный список extcommunity-list. extcommunity - это расширенная ценность сообщества. Внешнее сообщество компилируется в расширенную структуру сообщества. Мы можем определить несколько extcommunity-list под одним и тем же именем. В этом случае совпадение произойдет в порядке, определенном пользователем. Как только extcommunity-list совпадает с атрибутом extended communities в обновлениях BGP, он возвращает разрешение или запрет на основе определения extcommunity-list. Если нет совпадающей записи, будет возвращено значение deny. Когда extcommunity пуст, он соответствует любым маршрутам.

Применяется специальная обработка для internetсообщества. Он подходит для любого сообщества.

bgp extcommunity-list expandedNAMEpermit|denyLINE

Эта команда определяет новый расширенный список extcommunity-list. строка - это строковое выражение атрибута расширенных сообществ. строка может быть регулярным выражением (регулярные выражения BGP), чтобы соответствовать расширенному атрибуту сообществ в обновлениях BGP.

Обратите внимание, что все расширенные списки сообщества используют одно пространство имен, поэтому нет необходимости указывать их тип при их создании или уничтожении.

show bgp extcommunity-list[NAME detail]

Эта команда отображает текущую информацию о extcommunity-list. При указании имени отображается информация о списке сообщества.

1.8.4.4.21.8 Расширенные сообщества BGP на карте маршрутов

match extcommunityWORD

set extcommunity none

Эта команда сбрасывает значение расширенного сообщества в обновлениях BGP. Если атрибут уже настроен или получен от однорангового узла, атрибут отбрасывается и устанавливается в значение none. Это полезно, если вам нужно удалить входящие расширенные сообщества.

set extcommunity rt EXTCOMMUNITY

Эта команда задает целевое значение маршрута.

set extcommunity soo EXTCOMMUNITY

Эта команда задает значение исходного сайта.

set extcommunitybandwidth <(1-25600) | cumulative | num-multipaths> [non-transitive]

Эта команда устанавливает расширенное сообщество BGP link-bandwidth для префикса (наилучшего пути), для которого он применяется. Пропускная способность канала может быть указана как explicit value (указывается в Мбит/с), или маршрутизатору может быть предложено использовать cumulative bandwidth всех многолучевых путей для префикса или для его вычисления на основе number of multipaths. Сообщество с расширенной пропускной способностью канала кодируется как transitive если команда set явно не настраивает ее как non-transitive.

Обратите внимание, что расширенное расширенное сообщество используется только для совпадения правило, не для установить Действия.

1.8.4.4.21.9 Атрибут крупных сообществ

Атрибут крупных сообществ BGP был введен в феврале 2017 года с RFC 8092.

Атрибут больших сообществ BGP похож на атрибут сообществ BGP, за исключением того, что он содержит 3 компонента вместо двух и каждый из которых имеет длину 4 октета. Большие сообщества предоставляют дополнительную функциональность и удобство по сравнению с традиционными сообществами, в частности, тот факт, что GLOBAL теперь приведенная ниже часть имеет ширину 4 октета, что позволяет беспрепятственно использовать ее в сетях, использующих 4-байтовые ASN.

GLOBAL:LOCAL1:LOCAL2

Это формат для определения ценностей большого сообщества. Ссылаясь на RFC 8195, значения обычно обозначаются следующим образом:

GLOBAL Часть представляет собой поле глобального администратора 4 октета, обычно используемое в качестве операторов в КАЧЕСТВЕ номера.

LOCAL1 Часть представляет собой подполе локальных данных 4 октета, часть 1, называемое функцией.

LOCAL2 Часть представляет собой поле локальных данных 4 октета, часть 2, и называется подполем параметра.

В качестве примера, 65551:1:10 представляет КАК 65551 функцию 1 и параметр 10. Приведенный выше RFC дает некоторые рекомендации по рекомендуемому использованию.

1.8.4.4.22 Роли BGP

Роли BGP определены в RFC 9234 и обеспечивают простой способ предотвращения, обнаружения и устранения утечек.

Чтобы включить его механизм, вы должны настроить свою локальную роль так, чтобы она отражала ваш тип пиринговых отношений с вашим соседом. Возможные значения LOCAL-ROLE:

provider
rs-server
rs-client
customer
peer

Значение локальной роли согласовывается с новой возможностью роли BGP со встроенной проверкой соответствующего значения. В случае несоответствия будет отправлено уведомление о несоответствии новых ОТКРЫТЫХ ролей <2, 11>.

Правильные пары ролей:

Provider - Customer

Peer - Peer

RS-Server - RS-Client

```
~# vtysh -c 'show bgp neighbor' | grep'Role'
  Local : customer
   NeighborRole: provider
    Role: advertised and received
```

Если установлен строгий режим, сеанс BGP не будет установлен, пока сосед BGP не установит локальную роль на своей стороне. Этот параметр конфигурации определен в RFC 9234 и используется для принудительного выполнения соответствующей конфигурации на стороне вашей противоположной части. Значение по умолчанию - отключено.

Маршруты, отправленные от поставщика, rs-сервера или локальной роли партнера (или, если они получены клиентом, rs-clinet или локальной ролью партнера), будут помечены новым атрибутом только для клиента (OTC).

Маршруты с этим атрибутом могут быть отправлены вашему соседу, только если ваша локальная роль - provider или rs-server. Маршруты с этим атрибутом могут быть получены, только если ваша локальная роль - customer или rs-client.

В случае одноранговых отношений маршруты могут быть получены, только если значение OTC равно вашему соседу в качестве номера.

Все эти правила с OTC помогают обнаруживать и устранять утечки маршрутов и происходят автоматически, если установлена локальная роль.

neighbor PEER local-role LOCAL-ROLE[strict-mode]

Эта команда устанавливает вашу локальную роль в LOCAL-ROLE: <поставщик | rs-сервер | rs-клиент | клиент | одноранговый узел>.

Эта роль помогает обнаруживать и предотвращать утечки маршрутов.

Если strict-mode если установлено, ваш сосед должен отправить вам возможность со значением своей роли (установив local-role на его стороне). В противном случае будет отправлено уведомление о несоответствии ролей.

1.8.4.4.23 L3VPN VRFs

bgpd поддерживает L3VPN VRFs для IPv4 **RFC 4364** и IPv6 **RFC 4659**. Маршруты L3VPN и связанные с ними метки VRF MPLS могут быть распространены среди соседей VPN SAFI в *По умолчанию*, т.е. не VRF, экземпляр BGP. Метки VRF MPLS достигаются с помощью *Ядро* Метки MPLS, которые распространяются с использованием LDP или BGP, помечены как одноадресные. *bgpd* также поддерживается утечка между VRF-маршрутами.

1.8.4.4.23.1 L3VPN через интерфейсы GRE

В MPLS-VPN или SRv6-VPN для записи следующего перехода L3VPN требуется, чтобы выбранный путь соответственно содержал помеченный путь или действительный IPv6-адрес SID. В противном случае запись L3VPN не будет установлена. Можно проигнорировать эту проверку, если путь, выбранный следующим переходом, использует интерфейс GRE, и на входящей стороне семейства адресов ipv4-vpn или ipv6-vpn настроена карта маршрутов со следующим синтаксисом:

set l3vpn next-hopencapsulationgre

Входящая запись BGP L3VPN принимается при условии, что следующий переход записи L3VPN использует путь, который использует туннель GRE в качестве исходящего интерфейса. Удаленная конечная точка должна быть настроена сразу за туннелем GRE; конфигурация удаленного устройства может меняться в зависимости от того, работает ли оно на пограничной конечной точке или нет: в любом случае ожидается, что входящий трафик MPLS, полученный на этой конечной точке, должен рассматриваться как допустимый путь для L3VPN.

1.8.4.4.24 Утечка маршрута VRF

Маршруты BGP могут быть пропущены (т.Е. скопированы) между одноадресным VRF-каналом и VPN SAFI-каналом VRF по умолчанию для использования в L3VPN на основе MPLS. Одноадресные маршруты также могут передаваться между любыми VRFS (включая одноадресный RIB экземпляра BGP по умолчанию). Также доступен синтаксис быстрого доступа для указания утечки из одного VRF в другой VRF с использованием VPN RIB экземпляра по умолчанию в качестве посредника. Обычным применением функции VRF-VRF является подключение частного домена маршрутизации клиента к VPN-сервису провайдера. Утечка настраивается с точки зрения отдельного VRF: `import` относится к утечке маршрутов из VPN в одноадресный VRF, тогда как `export` относится к утечке маршрутов из одноадресной VRF в VPN.

1.8.4.4.24.1 Требуемые параметры

Маршруты, экспортируемые из одноадресного VRF в VPN RIB, должны быть дополнены двумя параметрами:

RD

RTLIST

Конфигурация для этих экспортированных маршрутов должна, как минимум, указывать эти два параметра.

Маршруты, импортируемые из VPN RIB в одноадресный VRF, выбираются в соответствии с их списками RTL. Утечка маршрутов, список RTL которых содержит по крайней мере одну цель маршрута, общую с настроенным списком RTL импорта. В конфигурации для этих импортированных маршрутов должен быть указан список RTL для сопоставления.

RD, не имеющая семантической ценности, предназначена для того, чтобы сделать маршрут уникальным в VPN RIB среди всех маршрутов с его префиксом, которые исходят от всех клиентов и сайтов, подключенных к VPN-сервису провайдера. Соответственно, каждому сайту каждого клиента обычно назначается удаленный сервер, который является уникальным во всей сети провайдера.

RTLIST - это набор значений расширенного сообщества, предназначенных для маршрута, целью которых является определение политики утечки маршрута. Как правило, клиенту присваивается одно целевое значение маршрута для импорта и экспорта, которое будет использоваться на всех сайтах клиентов. Эта конфигурация определяет простую топологию, в которой у клиента есть один домен маршрутизации, который является общим для всех его сайтов. Более сложные топологии маршрутизации возможны благодаря использованию дополнительных целевых объектов маршрута для увеличения утечки наборов маршрутов различными способами.

При использовании синтаксиса быстрого доступа для утечки из vrf в vrf, RD и RT выводятся автоматически.

1.8.4.4.24.2 Общая конфигурация

Настройка утечки маршрута между одноадресным VRF-каналом и VPN SAFI-каналом VRF по умолчанию выполняется с помощью команд в контексте семейства адресов VRF:

rd vpn export AS:NN|IP:nn

Указывает различитель маршрута, который должен быть добавлен к маршруту, экспортированному из текущего одноадресного VRF в VPN.

rt vpn import|export|both RTLIST...

Указывает список целевых маршрутов, который должен быть присоединен к маршруту (экспорт), или список целевых маршрутов для сопоставления (импорт) при экспорте / импорте между текущим одноадресным VRF и VPN.

Список RTLIST представляет собой разделенный пробелом список целевых объектов маршрута, которые являются значениями расширенного сообщества BGP, как описано в Атрибут расширенных сообществ.

label vpn export(0..1048575)|auto

Позволяет прикрепить метку MPLS к маршруту, экспортированному из текущего одноадресного VRF в VPN. Если указанное значение равно `auto` значение метки автоматически присваивается из пула, поддерживаемого демоном Zebra. Если Zebra не запущена или если эта команда не настроена, автоматическое присвоение метки не завершится, что заблокирует соответствующий экспорт маршрута.

nexthop vpn export A.B.C.D|X:X::X:X

Задает необязательное значение nexthop, которое должно быть присвоено маршруту, экспортированному из текущего одноадресного VRF в VPN. Если не указано, значение nexthop будет установлено на 0.0.0.0 или 0:0::0:0 (self).

route-map vpn import|export MAP

Задает необязательную карту маршрутов, которая будет применяться к маршрутам, импортируемым или экспортируемым между текущим одноадресным VRF и VPN.

import|export vpn

Позволяет импортировать или экспортировать маршруты между текущим одноадресным VRF и VPN.

import vrf VRFNAME

Синтаксис быстрого доступа для указания автоматической утечки из vrf VRFNAME в текущий VRF с использованием VPN RIB в качестве посредника. RD и RT являются автоматическими и не должны указываться явно ни для исходного, ни для целевого VRF.

Этот синтаксический режим быстрого доступа несовместим с явными инструкциями import vpn и export vpn для двух задействованных VRF. CLI запретит попытки настроить несовместимые режимы утечки.

bgp retain route-target all

Можно сохранить или нет префиксы VPN, которые не импортируются локальной конфигурацией VRF. Это можно сделать с помощью следующей команды в контексте глобального семейства VPNv4 / VPNv6. По умолчанию эта команда включена и не отображается. Отображается форма команды *no bgp retain route-target all*.

neighbor <A.B.C.D|X:X::X:X|WORD> soo EXTCOMMUNITY

Без этой команды атрибут расширенного сообщества SoO настраивается с использованием карты входящих маршрутов, которая устанавливает значение SoO в процессе обновления. С введением новой функции BGP для каждого соседнего сайта происхождения (SoO) две новые команды, настроенные в подрежимах в режиме конфигурации маршрутизатора, упрощают настройку значения SoO.

Если мы настроим SoO для каждого соседа в PEs, сообщество SoO автоматически добавляется для всех маршрутов из CPE. Маршруты проверены и не могут быть отправлены обратно в тот же CPE (например, для нескольких сайтов). Это особенно необходимо при использовании as-overrideили allowas-inдля предотвращения циклов маршрутизации.

mpls bgp forwarding

Можно разрешить BGP устанавливать префиксы VPN без меток транспорта, выполнив следующую команду в контексте конфигурации интерфейса. В этой конфигурации будут установлены префиксы VPN, созданные в сеансе e-bgp, и с прямым подключением к следующему переходу.

1.8.4.4.25 L3VPN SRv6

segment-routing srv6

Используйте серверную часть SRv6 с BGP L3VPN и перейдите к его узлу конфигурации.

locator

Укажите локатор SRv6, который будет использоваться для SRv6 L3VPN. Имя локатора должно быть задано в zebra, но пользователь может задать его в любом порядке.

1.8.4.4.25.1 Общая конфигурация

Настройка SID SRv6, используемого для объявления L3VPN как для IPv4, так и для IPv6, выполняется с помощью следующей команды в контексте VRF:

sid vpn per-vrf экспорт (1..1048575)|авто

Позволяет привязать SID SRv6 к маршруту, экспортированному из текущего одноадресного VRF в VPN. Один SID используется как для семейств адресов IPv4, так и для IPv6. Если вы хотите установить SID только для семейства адресов IPv4 или семейства адресов IPv6, вам необходимо использовать команду `sid vpn export (1..1048575)|auto` в контексте семейства адресов. Если указанное значение равно `auto` значение SID автоматически

присваивается из пула, поддерживаемого демоном Zebra. Если Zebra не запущена или если эта команда не настроена, автоматическое назначение идентификатора SID не завершится, что заблокирует соответствующий экспорт маршрута.

1.8.4.4.26 Виртуальная сеть Ethernet – EVPN

Примечание: При использовании функций EVPN и при наличии большого количества хостов обязательно отрегулируйте размер кэша соседей arp, чтобы избежать переполнения соседней таблицы и / или чрезмерной сборки мусора. В Linux размер таблицы и частоту сборки мусора можно регулировать с помощью следующих конфигураций sysctl:

```
net.ipv4.neigh.default.gc_thresh1 net.ipv4.neigh.default.gc_thresh2
net.ipv4.neigh.default.gc_thresh3 net.ipv6.neigh.default.gc_thresh1
net.ipv6.neigh.default.gc_thresh2 net.ipv6.neigh.default.gc_thresh3
```

Для получения дополнительной информации см. `man 7 arp`.

1.8.4.4.26.1 Включение EVPN

EVPN должен быть включен на экземпляре BGP, соответствующем VRF, действующему в качестве основы для туннелирования VXLAN. В большинстве случаев это будет VRF по умолчанию. Команда для включения EVPN для экземпляра BGP находится `advertise-all-vni` в `address-family l2vpn evpn`:

```
router bgp 65001
 !
 address-family l2vpn evpn
 advertise-all-vni
```

Более полный пример конфигурации можно найти в EVPN.

1.8.4.4.26.2 Цели маршрута EVPN L3

route-target <import|export|both> <RTLIST|auto>

Измените целевой набор маршрутов для объявленных EVPN маршрутов типа 2 / типа 5. RTLIST - это список любого из совпадений `(A.B.C.D:MN|EF:OPQR|GHJK:MN|*:OPQR|*:MN)`, где `*` указывается соответствие подстановочных знаков для номера AS. Он будет установлен в соответствии с любым номером AS. Это полезно при развертывании центров обработки данных с нижестоящими VNI. `auto` используется для сохранения автоконфигурации, которая является поведением по умолчанию для RTS L3.

1.8.4.4.26.3 EVPN advertise-PIP

При развертывании MLAG симметричной маршрутизации EVPN все маршруты EVPN объявляются с использованием anycast-IP в качестве следующего IP-адреса и anycast MAC в качестве MAC-адреса маршрутизатора (RMAC - в расширенном сообществе BGP EVPN). EVPN получает IP-адрес следующего перехода из локального туннельного IP-адреса интерфейса VxLAN, а RMAC получается из MAC-адреса SVI-интерфейса L3VNI. Примечание: IP-адрес следующего перехода используется для маршрутов EVPN независимо от того, развернута симметричная маршрутизация или нет, но RMAC имеет значение только для сценария симметричной маршрутизации.

Текущее поведение не является идеальным для префиксных (тип-5) и самостоятельных (тип-2) маршрутов. Это связано с тем, что трафик от удаленных VTEP маршрутизируется неоптимально, если они попадают в систему, которой маршрут не принадлежит.

Функция advertise-pip рекламирует префиксные (тип-5) и собственные (тип-2) маршруты с индивидуальным (основным) системным IP в качестве следующего перехода и отдельного

(системного) MAC в качестве MAC-адреса маршрутизатора (RMAC), оставляя поведение неизменным для других маршрутов EVPN.

Для поддержки этой функции необходимо иметь возможность сосуществования пары (system-MAC, system-IP) с парой (anycast-MAC, anycast-IP) с возможностью завершения пакетов, инкапсулированных в VxLAN, полученных для любой пары в одном и том же L3VNI (т.е. Связанной VLAN). Эта возможность необходима для каждого экземпляра VRF клиента.

Для получения системного MAC и anycast MAC должен быть отдельный / дополнительный интерфейс MAC-VLAN, соответствующий SVI L3VNI. MAC-адрес интерфейса SVI может быть интерпретирован как system-MAC, а MAC-адрес интерфейса MAC-VLAN как anycast MAC.

Для получения системных IP и anycast-IP идентификатор маршрутизатора экземпляра BGP по умолчанию используется в качестве system-IP, а IP-адрес локального туннеля интерфейса VxLAN - в качестве anycast-IP.

Пользователь может настроить значение system-IP и / или system-MAC, если значение, полученное автоматически, не является предпочтительным.

Примечание: По умолчанию функция advertise-pip включена, и у пользователя есть возможность отключить эту функцию через интерфейс командной строки конфигурации. Если функция отключена в экземпляре bgp vrf или интерфейс MAC-VLAN не настроен, все маршруты ведут себя одинаково, используя одинаковые значения следующего перехода и RMAC.

advertise-pip [ip <addr> [mac <addr>]]

Включает или отключает функцию рекламы-pip, указав параметры system-IP и / или system-MAC.

1.8.4.4.26.4 EVPN advertise-svi-ip

Как правило, IP-адрес SVI повторно используется в VTEPs на нескольких стойках. Однако, если у вас есть уникальные IP-адреса SVI, которые вы хотите сделать доступными, вы можете использовать опцию advertise-svi-ip. Этот параметр объявляет IP / MAC-адрес SVI как маршрут типа 2 и устраняет необходимость в любом потоке по VXLAN для достижения IP-адреса с удаленного VTEP.

advertise-svi-ip

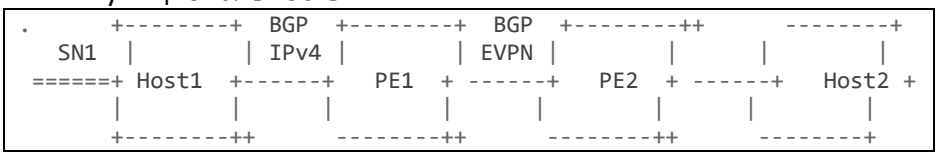
Обратите внимание, что не следует одновременно включать advertise-svi-ip и advertise-default-gw.

1.8.4.4.26.5 EVPN Overlay Index Gateway IP

RFC https://datatracker.ietf.org/doc/html/rfc9136 объясняет использование индексов наложения для рекурсивного разрешения маршрута для маршрута EVPN типа 5.

Мы поддерживаем индекс наложения IP-адресов шлюза. IP-адрес шлюза, объявленный с префиксом маршрута EVPN, используется для поиска маршрута EVPN MAC / IP с таким же полем IP, что и IP-адрес шлюза. Эта запись MAC / IP предоставляет VTEP nexthop и информацию о туннеле, необходимую для инкапсуляции VxLAN.

Функциональность:



Рассмотрим топологию выше, в которой префикс SN1 подключен за host1. Host1 объявляет SN1 PE1 через сеанс BGP IPv4. PE1 объявляет SN1 PE2, используя маршрут EVPN типа 5 с IP-адресом host1 в качестве IP-адреса шлюза. PE1 также объявляет MAC / IP Host1 как маршрут типа 2, который используется для разрешения IP-адреса шлюза host1.

PE2 получает этот маршрут типа 5 и импортирует его в vrf на основе целевых объектов маршрута. Префикс BGP, импортированный в vrf, использует IP-адрес шлюза в качестве BGP nexthop. Этот маршрут устанавливается в zebra, если выполняются следующие условия:

1. Следующий IP-адрес шлюза доступен на уровне L3.
2. PE2 получил маршрут EVPN типа 2 с полем IP, установленным на IP шлюза.

Требования к топологии:

1. Эта функция поддерживается только для асимметричной модели маршрутизации. При отправке пакетов в SN1 входящий PE (PE2) выполняет маршрутизацию, а выход PE (PE1) выполняет только соединение.
2. Эта функция поддерживает только традиционную (не поддерживающую vlan) модель моста. Интерфейс моста, связанный с L2VNI, является интерфейсом L3. т.Е. Этот интерфейс настроен с адресом в подсети L2VNI. Обратите внимание, что IP-адрес шлюза также должен иметь адрес в той же подсети.
3. Поскольку эта функция работает в асимметричной модели маршрутизации, все L2VNI и соответствующие интерфейсы VxLAN и bridge должны присутствовать во всех PE.
4. Для создания и импорта маршрутов EVPN типа 5 требуется конфигурация L3VNI. Также должны присутствовать интерфейсы L3VNI VxLAN и bridge.

PE может использовать один из следующих двух механизмов для объявления маршрута EVPN типа 5 с IP-адресом шлюза.

1. CLI для добавления IP-адреса шлюза при генерации маршрута EVPN типа 5 из префикса BGP IPv4 / IPv6:
   advertise <ipv4|ipv6> unicast [gateway-ip]

Когда этот интерфейс командной строки настроен для BGP vrf в семействе адресов L2VPN EVPN, маршруты EVPN типа 5 генерируются для префиксов BGP в vrf. Следующая точка префикса BGP становится IP-адресом шлюза соответствующего маршрута типа 5.

Если приведенная выше команда настроена без ключевого слова “gateway-ip”, маршруты типа 5 генерируются без оверлейного индекса.

Добавьте IP-адрес шлюза в маршрут EVPN type-5, используя карту маршрутов:
set evpn gateway-ip <ipv4|ipv6> <addr>

Когда карта маршрута с указанным выше предложением set применяется в качестве исходящей политики в BGP, он установит IP-адрес шлюза в EVPN type-5 NLRI.

Пример конфигурации:

```
router bgp 100
 neighbor  192.168.0.1 remote-as 101
 !
 address-family ipv4 l2vpn evpn
 neighbor  192.168.0.1 route-map RMAP
out
 exit-address-family
!
route-map10
 set evpn gateway-ip 10.0.0.1
 set evpn gateway-ip 10::1
```

У PE, который получает маршрут типа 5 с индексом наложения IP-адреса шлюза, должна быть включена конфигурация “разрешить разрешение наложения индекса”, чтобы рекурсивно разрешить индекс наложения nexthop и установить префикс в zebra.

enable-resolve-overlay-index

Пример конфигурации:

```
router bgp 65001
```

```
bgp router-id 192.168.100.1
no bgp ebgp-requires-policy
neighbor  10.0.1.2 remote-as 65002
!
address-family l2vpn evpn
neighbor  10.0.1.2 activate
advertise-all-vni
 enable-resolve-overlay-index
exit-address-family
!
```

1.8.4.4.26.6 EVPN Multihoming

Полностью активная многодомовая передача используется для резервирования и распределения нагрузки. Серверы подключены к двум или более PE, и ссылки связаны (объединение ссылок). Эта группа серверных соединений называется сегментом Ethernet.

1.8.4.4.26.6.1 Сегменты Ethernet

Сегмент Ethernet можно настроить, указав системный MAC-адрес и локальный дискриминатор или полное ESINAME для интерфейса связи на PE (через zebra) –

evpn mh es-id <(1-16777215)|ESINAME>

evpn mh es-sys-mac X:X:X:X:X:X

Sys-mac и локальный дискриминатор используются для генерации 10-байтового идентификатора сегмента Ethernet 3-го типа. ESINAME - это 10-байтовый идентификатор сегмента Ethernet типа 0 - “00:AA: BB: CC: DD: EE: FF: GG: HH: II”.

Маршруты типа 1 (EAD-per-ES и EAD-per-EVI) используются для объявления локально подключенных ESS и для обучения удаленных ESS в сети. Локальные маршруты типа 2 / MAC-IP также рекламируются с помощью целевого ESI, позволяющего синхронизировать MAC-IP между одноранговыми узлами сегмента Ethernet. Ссылка: RFC 7432, RFC 8365

EVPN-MH предназначен для замены MLAG или Anycast VTEPs. В multihoming каждый PE имеет уникальный адрес VTEP, который требует введения новой конструкции dataplane, MAC-ECMP. Здесь запись MAC / FDB может указывать на список удаленных PE / VTEP.

1.8.4.4.26.6.2 Обработка BUM

Маршруты типа 4 (ESR) используются для выбора назначенного экспедитора (DF). DFs перенаправляет трафик BUM, полученный через оверлейную сеть. В этой реализации используется выбор DF на основе предпочтений, указанный в draft-ietf-bess-evpn-pref-df. Параметр DF настраивается индивидуально (через zebra) –

evpn mh es-df-pref (1-16777215)

Трафик BUM перенаправляется через оверлей всеми PE, подключенными к серверу, но только DF может перенаправлять декапсулированный трафик на порт доступа. Чтобы учесть, что в плане данных установлены фильтры, отличные от DF, для отбрасывания трафика.

Аналогично трафик, полученный от одноранговых узлов ES через оверлей, не может быть перенаправлен на сервер. Это фильтрация с разделением горизонта с локальным смещением.

1.8.4.4.26.6.3 Кнопки для взаимодействия

Некоторые поставщики не отправляют маршруты EAD-per-EVI. Для взаимодействия с ними нам нужно ослабить зависимость от маршрутов EAD-per-EVI и активировать удаленный ES-PE на основе только маршрута EAD-per-ES.

Обратите внимание, что по умолчанию мы объявляем и ожидаем маршруты EAD-per-EVI.

disable-ead-evi-rx

disable-ead-evi-tx

1.8.4.4.26.6.4 Быстрый переход на другой ресурс

Поскольку основной целью EVPN-MH является резервирование, поддержание эффективности отработки отказа является повторяющейся темой в реализации. Следующие дополнительные функции были введены специально для обеспечения эффективного восстановления работоспособности ES.

Группы Nexthop уровня 2 и MAC-ECMP через L2NHG.

Маршруты хоста (для симметричного IRB) через L3NHG. На плоскостях данных, поддерживающих layer3 nexthop groups, эту функцию можно включить с помощью следующей конфигурации BGP –

use-es-l3nhg

Переключение на локальную ES (MAC / Neigh) с помощью ES-redirect. На плоскостях данных, которые не поддерживают ES-перенаправление, эту функцию можно отключить с помощью следующей конфигурации zebra –

evpn mh redirect-off

1.8.4.4.26.6.5 Восходящий канал / отслеживание ядра

Когда все нижележащие ссылки отключаются, PE больше не имеет доступа к наложению VxLAN +. Чтобы предотвратить утечку трафика, ссылки на сервер / ES протодиализируются в PE. Связь может быть настроена для отслеживания восходящей линии связи с помощью следующей конфигурации zebra –

evpn mh uplink

1.8.4.4.26.6.6 Реклама прокси

Для обработки обновлений без сбоев добавлена поддержка прокси-рекламы, как указано в draft-rbickhart-evpn-ip-mac-proxy-adv. Это позволяет PE (скажем, PE1) через прокси-сервер рекламировать MAC-IP, rxed от одноранговой узла ES (скажем, PE2). Когда одноранговый узел ES (PE2) выходит из строя, PE1 продолжает рекламировать узлы, полученные от PE2, на время ожидания, в течение которого он пытается установить локальную доступность узла. Это время ожидания настраивается с помощью следующих команд zebra –

evpn mh neigh-holdtime (0-86400)

evpn mh mac-holdtime (0-86400)

1.8.4.4.26.6.7 Задержка запуска

Когда коммутатор перезагружается, мы ждем в течение короткого периода времени, чтобы позволить базовой сети и EVPN сойтись, прежде чем включить ESs. В течение этого срока облигации ES хранятся в протодауне. Задержка запуска настраивается с помощью следующей команды zebra –

evpn mh startup-delay(0-3600)

1.8.4.4.26.6.8 Фрагментация EAD-per-ES

Маршрут EAD-per-ES содержит цели маршрута EVI для всех широковещательных доменов, связанных с ES. В зависимости от масштаба EVI маршрут EAD-per-ES может быть фрагментирован.

Количество EVI для каждого маршрута EAD может быть настроено с помощью следующей команды BGP –

[no] ead-es-frag evi-limit (1-1000)

Пример конфигурации

```
!
! bgp 5556
 !
 address-family l2vpn evpn
 ead-es-frag evi-limit 200
 exit-address-family
 !
!
```

1.8.4.4.27 Поддержка с помощью серверной части сетевого пространства имен VRF

С помощью VRFs можно отделить оверлейные сети, содержащиеся в интерфейсах VXLAN, от нижележащих сетей. Для этого можно использовать бэкенды VRF-lite и VRF-netns. В последнем случае необходимо установить как мост, так и интерфейс vxlan в одном и том же пространстве имен сети, как показано в приведенном ниже примере:

```
# оболочка linux ip netns Добавить vrf1 ip Ссылка Добавить Имя vxlan101 Тип vxlan ID 101 dstport
4789 разработчик eth0 Местные новости  10.1.1.1
  .1.1.1 iplinkset разработчик vxlan101 netns vrf1 ip netns exec  vrf1ip Ссылка установить
разработчик lo up ip netns exec vrf1 brctl addbr ip-адрес bridge101 netns exec vrf1 brctl addif
bridge101 vxlan101
```

Это позволяет разделять не только сети уровня 3, такие как сети VRF-lite. Кроме того, сети VRF на основе позволяют разделить сети уровня 2 на отдельные экземпляры VRF.

1.8.4.4.28 Условная реклама BGP

Функция условной рекламы BGP использует `non-exist-map` или `exist-map` и `advertise-map` ключевые слова команды neighbor advertise-map для отслеживания маршрутов по префиксу маршрута.

`non-exist-map`

1. Если префикс маршрута отсутствует в выходных данных команды несуществующей карты, затем объявите маршрут, указанный командой advertise-map .
2. Если префикс маршрута присутствует в выходных данных команды несуществующей карты, то не рекламируйте маршрут, указанный командой addvertise-map .

`exist-map`

1. Если в выходных данных команды exist-map присутствует префикс маршрута, то объявите маршрут, указанный командой advertise-map.
2. Если префикс маршрута отсутствует в выходных данных команды exist-map, не рекламируйте маршрут, указанный командой advertise-map .

Эта функция полезна, когда некоторые префиксы объявляются одному из его одноранговых узлов, только если информация от другого узла отсутствует (из-за сбоя в сеансе пиринга или частичной доступности и т. Д.).

Условные объявления BGP отправляются в дополнение к обычным объявлениям, которые маршрутизатор BGP отправляет своему одноранговому узлу.

Процесс условной рекламы запускается процессом сканирования BGP, который по умолчанию выполняется каждые 60. Это означает, что максимальное время вступления в силу условного объявления равно значению таймера процесса.

В качестве оптимизации, хотя процесс всегда выполняется по истечении каждого таймера, он определяет, изменилась ли политика условной рекламы или таблица маршрутизации; если ни то, ни другое не изменилось, обработка не требуется, и сканер завершается досрочно.

neighbor A.B.C.D advertise-map NAME[exist-map|non-exist-map] NAME

Эта команда позволяет процессу сканирования BGP отслеживать маршруты, указанные командой exist-map или не-exist-map в таблице BGP, и условно объявляет маршруты, указанные командой advertise-map.

bgp (5-240)

Установите период для повторного запуска процесса проверки условной рекламы. Значение по умолчанию равно 60 секундам.

Пример конфигурации

```
interface enp0s9
 ip address 10.10.10.2/24
!
interface enp0s10
 ip address
10.10.20.2/24!
10.10.20.2/ 24!ip address 203.0.113.1/32
!
203.0.113.1/32!2
 bgp2no bgp ebgp-requires-policy
 neighbor 10.10.10.1 remote-as  10.10.10.1
 neighbor 1 remote-as  10.10.20.3
 !
 3 !neighbor 10.10.10.1 soft-reconfiguration inbound
  10.10.10.1 10.10.20.3 soft-reconfiguration inbound
  10.10.20.3 10.10.20.3 advertise-map ADV-MAP non-exist-map EXIST-MAP
 exit-address-family
!
ip prefix-list DEFAULT seq 5 permit 192.0.2.5/32
ip prefix-list DEFAULT seq 10 permit 192.0.2.1/32
ip prefix-list EXIST seq 5 permit 10.10.10.10/32
ip prefix-list DEFAULT-ROUTE seq 5 permit 0.0.0.0/0
ip prefix-list IP1 seq 5 permit 10.139.224.0/20
!
bgp community-list standard DC-ROUTES seq 5 permit 64952:3008
bgp community-list standard DC-ROUTES seq 10 permit 64671:501
bgp community-list standard DC-ROUTES seq 15 permit 64950:3009
bgp community-list standard DEFAULT-ROUTE seq 5 permit 65013:200
!
route-map ADV-MAP permit 10
 match ip address prefix-list IP1

!
route-map ADV-MAP permit 20
 match community DC-ROUTES
!
route-map EXIST-MAP permit 10
```

```
 match community DEFAULT-ROUTE
 match ip address prefix-list DEFAULT-ROUTE

!
```

1.8.4.4.28.1 Пример вывода

Когда маршрут по умолчанию присутствует в таблице BGP R2’2, 10.139.224.0/20 и 192.0.2.1/32 не объявляются R3.

```
Router2# show ip bgp
BGP table version is 20, local router ID is  , vrf id 0
Default local pref 100, local AS 2
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
 i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

 Network Next Hop Metric LocPrf Weight Path
*> 0.0.0.0/0         10.10.10.1               0             0  1 i
*> 10.139.224.0/20  10.10.10.1              0             0 1 ?
*> 192.0.2.1/32      10.10.10.1               0             0  1 i
*> 192.0.2.5/32     1               0             0 10.10.10.1 i

Displayed   4 routes and 4 total paths
Router2# show ip bgp neighbors 10.10.20.3

!--- Output suppressed.

For address family: IPv4 Unicast
Update 7, subgroup 7
Packet 0
Inbound soft reconfiguration allowed
Community attribute sent to this neighbor(all)
Condition NON_EXIST, Condition-map *EXIST-MAP, Advertise-map *ADV-MAP, status: Withdraw
0 accepted prefixes

!--- Output suppressed.

Router2# show ip bgp neighbors 10.10.20.3 advertised-routes
BGP table version is 20, local router ID is  , vrf id 0
Default local pref 100, local AS 2
 codes: s suppressed, d damped, h history, * valid, > best, = multipath,
 i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

 Network Next Hop Metric LocPrf Weight Path
*> 0.0.0.0/0         0.0.0.0                                0  1 i
*> 192.0.2.5/32      0.0.0.0                                0  1 i

Total number of prefixes 2
```

Если маршрут по умолчанию отсутствует в таблице BGP R2’2, 10.139.224.0/20 и 192.0.2.1/32 объявляются в R3.

```
Router2# show ip bgp
BGP table version is 21, local router ID is 203.0.113.1, vrf id 0
Default local pref 100, local AS 2
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop            Metric LocPrf Weight Path
```

```
*> 10.139.224.0/20  10.10.10.1               0             0 1 ?
*> 192.0.2.1/32     10.10.10.1               0             0 1 i
*> 192.0.2.5/32     10.10.10.1               0             0 1 i

Displayed  3 routes and 3 total paths

Router2# show ip bgp neighbors 10.10.20.3

!--- Output suppressed.

For address family: IPv4 Unicast
Update group 7, subgroup 7
Packet Queue length 0
Inbound soft reconfiguration allowed
Community attribute sent to this neighbor(all)
Condition NON_EXIST, Condition-map *EXIST-MAP, Advertise-map *ADV-MAP, status: Advertise
0 accepted prefixes

!--- Output suppressed.

Router2# show ip bgp neighbors 10.10.20.3 advertised-routes
BGP table version is 21, local router ID is 203.0.113.1, vrf id 0
Default local pref 100, local AS 2
Status codes:  s suppressed, d damped, h history, * valid, > best, = multipath,
               i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes:  i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.139.224.0/20  0.0.0.0                              0 1 ?
*> 192.0.2.1/32     0.0.0.0                              0 1 i
*> 192.0.2.5/32     0.0.0.0                              0 1 i

Total number of prefixes 3
Router2#
```

1.8.4.4.29 Отладка

show debug

Показать все разрешенные отладки.

show bgp

Отображение прослушиваемых сокетов и vrf, который их создал. Полезно для отладки, когда прослушивание не работает, и это считается инструкцией разработчика по отладке.

debug bgp allow-martian

Включить или отключить BGP, принимающий марсианские nexthops от однорангового узла. Пожалуйста, обратите внимание, что это не настоящая команда отладки, и эта команда также устарела и скоро будет удалена. Новая команда `bgp allow-martian-nexthop`

debug bgp bfd

Включить или отключить отладку для событий BFD. При этом будут показаны сообщения библиотеки интеграции BFD и сообщения интеграции BGP BFD, которые в основном связаны с переходами состояний и проблемами проверки.

debug bgp conditional-advertisement

Включить или отключить отладку условной рекламы BGP.

debug bgp neighbor-events

Включить или отключить отладку для соседних событий. Здесь представлена общая информация о событиях BGP, таких как одноранговое соединение / разъединение, установление / разрыв сеанса и согласование возможностей.

debug bgp updates

Включить или отключить отладку для обновлений BGP. Здесь представлена информация о сообщениях об ОБНОВЛЕНИИ BGP, передаваемых и принимаемых между локальным и удаленным экземплярами.

debug bgp keepalives

Включить или отключить отладку для BGP keepalives. Здесь представлена информация о сообщениях BGP KEEPALIVE, передаваемых и принимаемых между локальным и удаленным экземплярами.

debug bgp bestpath <A.B.C.D/M|X:X::X:X/M>

Включите или отключите отладку для выбора наилучшего пути для указанного префикса.

debug bgp nht

Включить или отключить отладку отслеживания BGP nexthop.

debug bgp update-groups

Включить или отключить отладку динамических групп обновлений. Здесь представлена общая информация о событиях создания, удаления, объединения и удаления групп.

debug bgp zebr

Включить или отключить отладку связи между *bgpd* и *zebra*.

1.8.4.4.29.1 Сброс сообщений и таблиц маршрутизации

dump bgp allPATH[INTERVAL]

dump bgp all-et PATH [INTERVAL]

Дамп всех пакетов и событий BGP в *путь*. Если *интервал* установлен, для echo будет создан новый файл *интервал* в секундах. Путь *путь* можно задать форматирование даты и времени (strftime). Тип ‘all-et’ включает поддержку расширенного заголовка метки времени (Формат двоичного дампа пакета).

dump bgp updatesPATH[INTERVAL]

dump bgp updates-et PATH [INTERVAL]

Сбрасывайте только сообщения об обновлениях BGP в файл *path*. Если задан *интервал*, новый файл будет создан для эхо-сигнала с *интервалом* в несколько секунд. Путь к *пути* можно задать с помощью форматирования даты и времени (strftime). Тип ‘updates-et’ включает поддержку расширенного заголовка временной метки (формат двоичного дампа пакета).

dump bgp routes-mrt PATH

dump bgp routes-mrt PATH INTERVAL

Дамп всей таблицы маршрутизации BGP в path. Это тяжелый процесс. Путь к пути можно задать с помощью форматирования даты и времени (strftime). Если задан интервал, новый файл будет создан для эхо-сигнала с интервалом в несколько секунд.

Примечание: переменная интервала также может быть установлена с использованием часов и минут: 04h20m00.

1.8.4.4.30 Другие команды BGP

В режиме *включения* верхнего уровня доступны следующие:

clear bgp \ *

Очистить все одноранговые узлы.

clear bgp ipv4|ipv6 \ *

Очистите все одноранговые узлы с активированным этим семейством адресов.

clear bgp ipv4|ipv6 *

Очистите все одноранговые узлы с активированным этим семейством адресов и семейством вложенных адресов.

clear bgp ipv4|ipv6 PEER

Очистить одноранговые узлы с адресом X.X.X.X и активировать это семейство адресов.

clear bgp ipv4|ipv6 unicast PEER

Очистить одноранговый узел с адресом X.X.X.X и активировать это семейство адресов и семейство вложенных адресов.

clear bgp ipv4|ipv6 PEER soft|in|out

Очистить одноранговый узел с помощью программной реконфигурации в этом семействе адресов.

clear bgp ipv4|ipv6 unicast PEER soft|in|out

Очистить одноранговый узел с помощью программной реконфигурации в этом семействе адресов и семействе вложенных адресов.

clear bgp [ipv4|ipv6] [unicast] PEER|* message-stats

Очистка статистики сообщений BGP для указанного однорангового узла или для всех одноранговых узлов, необязательно фильтруемых по активированному семейству адресов и семейству вложенных адресов.

В режиме `router bgp`:

write-quanta (1-64)

Ввод-вывод передачи сообщений BGP является векторным. Это означает, что несколько пакетов записываются в одноранговый сокет одновременно в каждом цикле ввода-вывода, чтобы минимизировать накладные расходы на системные вызовы. Это значение определяет, сколько записей записывается одновременно. При определенных условиях загрузки уменьшение этого значения может привести к снижению интенсивности однорангового трафика. На практике оставьте эти настройки по умолчанию (64), если вы действительно не знаете, что делаете.

read-quanta (1-10)

В отличие от Tx, трафик BGP Rx не является векторным. Пакеты считываются с провода по одному за раз в цикле. Этот параметр определяет, сколько итераций выполняется цикл. Как и в случае с квантами записи, лучше оставить этот параметр по умолчанию.

Следующая команда доступна как в `config` режиме, так и в `router bgp` режиме:

bgp graceful-shutdown

Цель этой команды - инициировать корректное завершение работы BGP, которое описано в RFC 8326. Это используется для минимизации или устранения потерь трафика в сети, когда на маршрутизаторе необходимо выполнить плановое техническое обслуживание, такое как обновление программного обеспечения или замена оборудования. Функция работает путем повторного объявления маршрутов для одноранговых узлов eBGP с включенным сообществом GRACEFUL_SHUTDOWN. Ожидается, что одноранговые узлы будут обрабатывать такие пути с наименьшим предпочтением. Это происходит автоматически на получателе, использующем FRR; с другими стеками протоколов маршрутизации, возможно, потребуется настроить входящую политику. В FRR запуск постепенного завершения работы также приводит к объявлению LOCAL_PREF 0 для одноранговых узлов iBGP.

Плавное завершение работы может быть настроено для каждого экземпляра BGP или глобально для всего BGP. Эти два варианта являются взаимоисключающими. Отсутствие формы команды приводит к остановке graceful shutdown, и маршруты будут повторно объявлены без сообщества GRACEFUL_SHUTDOWN и / или с обычным значением LOCAL_PREF. Обратите внимание, что если этот параметр сохранен в конфигурации запуска, постепенное завершение работы будет оставаться в силе при перезапусках *bgpd* и его необходимо будет явно отключить.

bgp input-queue-limit (1-4294967295)

Установите ограничение очереди ввода BGP для всех одноранговых узлов при анализе сообщений. Увеличивайте это значение, только если у вас достаточно памяти для одновременной обработки больших очередей сообщений.

bgp output-queue-limit (1-4294967295)

Установите ограничение очереди вывода BGP для всех одноранговых узлов при анализе сообщений. Увеличивайте это значение, только если у вас достаточно памяти для одновременной обработки больших очередей сообщений.

1.8.4.5 Отображение информации BGP

Следующие четыре команды отображают таблицы маршрутизации IPv6 и IPv4 в зависимости от того `ip`, используется ключевое слово или нет. На самом деле, `show ip bgp` command использовалась в более старом проекте *Quagga* routing daemon , в то время `show bgp` как command - это новый формат. Был сделан выбор в пользу сохранения старого формата с таблицей маршрутизации IPv4, в то время как новый формат отображает таблицу маршрутизации IPv6.

show ip bgp [all] [wide|json [detail]]

show ip bgp A.B.C.D [json]

show bgp [all] [wide|json [detail]]

show bgp X:X::X:X [json]

Эти команды отображают маршруты BGP. Если маршрут не указан, по умолчанию отображаются все маршруты BGP.

```
BGP table version is 0, local router ID is 10.1.1.1
 Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
 Origin codes: i - IGP, e - EGP, ? - incomplete

Network Next Hop Metric LocPrf Weight Path
 \*> 1.1.1.1/32 0.0.0.0 0 32768 i

 Total number of prefixes 1
```

Если **wide** опция указана, затем ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.

Это особенно удобно при работе с префиксами IPv6 и, если **[no] bgp default show-nexthop-hostname** включена.

Если **all** опция указана, **ip** ключевое слово игнорируется, команды show bgp all и show ip bgp all отображают маршруты для всех AFI и SAFI.

Если **json** опция указана, выходные данные отображаются в формате JSON.

Если **detail** опция указывается после **json** будет отображен более подробный вывод JSON.

Некоторые другие команды предоставляют дополнительные опции для фильтрации выходных данных.

show [ip] bgp

Эта команда отображает маршруты BGP, используя регулярное выражение AS path (регулярные выражения BGP).

show [ip] bgp [all] summary [wide] [json]

Показать сводку одноранговых узлов bgp для указанного семейства адресов.

Старая структура команд show ip bgpможет быть удалена в будущем и больше не должна использоваться. Для доступа к другим таблицам маршрутизации BGP, отличным от таблицы маршрутизации IPv6, предоставленнойshow bgp, новая структура команд расширена с show bgp [afi] [safi]помощью .

`wide` опция дает больше выходных данных, похожих `LocalAS` на и расширенных `Desc` до 64 символов.

```
exit1# show ip bgp summary wide

IPv4 Unicast Summary (VRF default):
BGP 192.168.100.1, local AS number 65534 vrf-id 0
BGP 3
RIB 5, using 920 bytes of memory
1 Peers 1, using 27 KiB of memory

Neighbor V AS LocalAS MsgRcvd MsgSent TblVer InQ OutQ Up/Down
State/PfxRcd PfxSnt Desc
192.168.0.2     4      65030        123        15        22
0    0    0 00:07:00            0        1 us-east1-
rs1.frrouting.org

Total number of neighbors 1
exit1#
```

show bgp [afi] [safi] [all] [wide|json]

show bgp vrfs [<VRFNAME$vrf_name>] [json]

Команда отображает базовую информацию обо всех экземплярах bgp vrf, такую как идентификатор маршрутизатора, настроенные и установленные соседи, базовую информацию,

связанную с evpn, такую как l3vni, router-mac, vxlan-интерфейс. Пользователь может получить эту информацию в формате JSON, когда `json` представлено ключевое слово в конце cli.

```
torc-11# show bgp vrfs
Type  Id     routerId          #PeersCfg  #PeersEstb  Name
             L3-VNI            RouterMAC
Interface
DFLT  0      17.0.0.6          3          3           default
             0                 00:00:00:00:00:00      unknown
 VRF  21     17.0.0.6          0          0           sym_1
             8888              34:11:12:22:22:01
vlan4034_l3
 VRF  32     17.0.0.6          0          0           sym_2
             8889              34:11:12:22:22:01
vlan4035_l3

Total number of VRFs (including default): 3
```

show bgp [<ipv4|ipv6> <unicast|multicast|vpn|labeled-unicast|flowspec> | l2vpn evpn]

Эти команды отображают маршруты BGP для конкретной таблицы маршрутизации, указанной выбранным afi и выбранным safi. Если не задано значение afi и safi, команда возвращается к таблице маршрутизации IPv6 по умолчанию.

show bgp l2vpn evpn route[type <macip|2|multicast|3|es|4|prefix|5>]

Префиксы EVPN также могут быть отфильтрованы по типу маршрута EVPN.

show bgp vni <all|VNI> [vtep VTEP] [type <ead|1|macip|2|multicast|3>] [<detail|json>]

Отображение таблицы маршрутизации для каждого VNI EVPN в bgp. Фильтровать по типу маршрута, vtep или VNI.

show bgp [afi] [safi] [all] summary [json]

Показать сводку одноранговых узлов bgp для указанного семейства адресов и последующего семейства адресов.

show bgp [afi] [safi] [all] summary failed [json]

Показать сводку одноранговых узлов bgp для одноранговых узлов, которые не успешно обмениваются маршрутами для указанного семейства адресов и последующего семейства адресов.

show bgp [afi] [safi] [all] summary established [json]

Покажите сводку одноранговых узлов bgp для одноранговых узлов, которые успешно обмениваются маршрутами для указанного семейства адресов и последующего семейства адресов.

show bgp [afi] [safi] [all] summary neighbor [PEER] [json]

Показать сводку bgp для указанного одноранговой узла, семейства адресов и последующего семейства адресов. Соседний фильтр можно использовать в сочетании с вышедшими из строя установленными фильтрами.

show bgp [afi] [safi] [all] summary remote-as <internal|external|ASN> [json]

Отображение сводки одноранговых узлов bgp для указанного удаленного as ASN или типа (`internal` для iBGP и `external` для сеансов eBGP), семейства адресов и последующего семейства адресов. Фильтр remote-as можно использовать в сочетании с неисправными установленными фильтрами.

show bgp [afi] [safi] [all] summary terse [json]

Сократите выход. Не отображается следующая информация об экземплярах BGP: количество записей RIB, версия таблицы и используемая память. Эта `terse` опция может использоваться в сочетании с фильтрами remote-as, neighbor, failed и established, а также с `wide` опцией.

show bgp [afi] [safi] [neighbor [PEER] [routes|advertised-routes|received-routes] [<A.B.C.D/M|X:X::X:X/M> | detail] [json]

Эта команда отображает информацию о конкретном узле BGP соответствующего выбранного afi и safi.

routesКлючевое слово отображает только маршруты в таблице BGP этого семейства адресов, которые были получены этим одноранговым узлом и приняты политикой входящего доступа.

advertised-routesКлючевое слово отображает только маршруты в таблице BGP этого семейства адресов, которые были разрешены политикой исходящих сообщений и объявлены этому узлу.

received-routesКлючевое слово отображает все маршруты, принадлежащие этому семейству адресов (до политики входящих), которые были получены этим одноранговым узлом.

Если указан конкретный префикс, будет отображена подробная версия этого префикса.

Если detailопция указана, будет отображена подробная версия всех маршрутов. Тот же формат, show [ip] bgp [afi] [safi] PREFIXкоторый будет использоваться, но для всей таблицы полученных, объявленных или отфильтрованных префиксов.

Если jsonопция указана, выходные данные отображаются в формате JSON.

show bgp [<view|vrf> VIEWVRFNAME] [afi] [safi] neighbors PEER received prefix-filter [json]

Отображать префикс адреса ORFs, полученный от этого одноранговой узла.

show bgp [afi] [safi] [all] dampening dampened-paths [wide|json]

Пути отображения подавлены из-за ослабления выбранного afi и выбранного safi.

show bgp [afi] [safi] [all] dampening flap-statistics [wide|json]

Отображение статистики маршрутов выбранных afi и выбранных safi

show bgp [afi] [safi] [all] dampening parameters [json]

Отображение сведений о настроенных параметрах демпфирования выбранных afi и safi.
Если jsonопция указана, выходные данные отображаются в формате JSON.

show bgp [afi] [safi] [all] version (1-4294967295) [wide|json]

Отображение префиксов с соответствующими номерами версий. Здесь будет указан номер версии и выше с префиксами.

Это помогает определить, какие префиксы были установлены в какой-то момент.

Вот пример того, как проверить, какие префиксы были установлены, начиная с произвольной версии:

```
# vtysh -c 'show bgp ipv4 unicast json' | jq '.tableVersion'
9
# vtysh -c 'show ip bgp version 9 json' | jq -r '.routes | keys[]'
192.168.3.0/24
# vtysh -c 'show ip bgp version 8 json' | jq -r '.routes | keys[]'
192.168.2.0/24
192.168.3.0/24
```

show bgp [afi] [safi] statistics

Отображение статистики маршрутов выбранных afi и safi.

show bgp statistics-all

Отображение статистики маршрутов всех afi и safi.

show [ip] bgp [afi] [safi] [all] cidr-only [wide|json]

Отображение маршрутов с ненатуральными сетевыми масками.

show [ip] bgp [afi] [safi] [all] prefix-list WORD [wide|json]

Отображение маршрутов, соответствующих указанному списку префиксов.
Если wideопция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.
Если jsonопция указана, выходные данные отображаются в формате JSON.

show [ip] bgp [afi] [safi] [all] access-list WORD [wide|json]

Отображение маршрутов, соответствующих указанному списку доступа.

show [ip] bgp [afi] [safi] [all] filter-list WORD [wide|json]

Отображение маршрутов, соответствующих указанному списку фильтров AS-Path.
Если wideопция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.
Если json опция указана, выходные данные отображаются в формате JSON.

show [ip] bgp [afi] [safi] [all] route-map WORD [wide|json]

Отображение маршрутов, соответствующих указанной карте маршрутов.
Если wideопция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.
Если jsonопция указана, выходные данные отображаются в формате JSON.

show [ip] bgp [afi] [safi] [all] <A.B.C.D/M|X:X::X:X/M> longer-prefixes [wide|json]

Отображает указанный маршрут и все более конкретные маршруты.

Если wide опция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.

Если jsonопция указана, выходные данные отображаются в формате JSON.

show [ip] bgp [afi] [safi] [all] neighbors A.B.C.D [advertised-routes|received-routes|filtered-routes] [<A.B.C.D/M|X:X::X:X/M> | detail] [json|wide]

Отображение маршрутов, объявленных соседу BGP, или полученных маршрутов от соседа, или отфильтрованных маршрутов, полученных от соседа, на основе указанного параметра.

Если wide опция указана, то ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.

Это особенно удобно при работе с префиксами IPv6 и если [no] bgp default show-nexthop-hostnameвключен.

Если allопция указана, ірключевое слово игнорируется и маршруты отображаются для всех AFI и SAFI. если afi указан с allпараметром, маршруты будут отображаться для каждого SAFI в выбранном AFI

Если указан конкретный префикс, будет отображена подробная версия этого префикса.

Если detailопция указана, будет отображена подробная версия всех маршрутов. Тот же формат, show [ip] bgp [afi] [safi] PREFIXкоторый будет использоваться, но для всей таблицы полученных, объявленных или отфильтрованных префиксов.

Если jsonопция указана, выходные данные отображаются в формате JSON.

show [ip] bgp [afi] [safi] [all] detail-routes

Отображение подробной версии всех маршрутов. Тот же формат, что и при использованииshow [ip] bgp [afi] [safi] PREFIX, но для всей таблицы BGP.

Если allопция указана, ірключевое слово игнорируется и маршруты отображаются для всех AFI и SAFI.

Если afiуказано с allпараметром, маршруты будут отображаться для каждого SAFI в выбранном AFI.

1.8.4.5.1 Отображение маршрутов по атрибуту сообщества

Следующие команды позволяют отображать маршруты на основе их атрибута сообщества.

show [ip] bgp <ipv4|ipv6> [all] community [wide|json]

show [ip] bgp <ipv4|ipv6> [all] community COMMUNITY [wide|json]

show [ip] bgp <ipv4|ipv6> [all] community COMMUNITYexact-match [wide|json]

Эти команды отображают маршруты BGP, которые имеют атрибут сообщества. атрибут. Когда `COMMUNITY` указан, отображаются маршруты BGP, соответствующие этому сообществу. Когда *точное соответствие* указан, он отображает только маршруты, которые имеют точное совпадение.

show [ip] bgp <ipv4|ipv6> community-list WORD[json]

show [ip] bgp <ipv4|ipv6> community-list WORDexact-match[json]

Эти команды отображают маршруты BGP для указанного семейства адресов, которые соответствуют указанному списку сообщества. Когда точное соответствие указан, он отображает только маршруты, которые имеют точное совпадение.

Если wide опция указана, затем ширина таблицы префиксов увеличивается, чтобы полностью отобразить префикс и следующую строку.

Это особенно удобно при работе с префиксами IPv6 и, если [no] bgp default show-nexthop-hostname включена.

Если all опция указана, ip ключевое слово игнорируется и маршруты отображаются для всех AFI и SAFI. если указан afi, с all опция, маршруты будут отображаться для каждого SAFI в выбранном AFI

Если json опция указана, выходные данные отображаются в формате JSON.

show bgp labelpool <chunks|inuse|ledger|requests|summary> [json]

Эти команды отображают информацию о пуле меток BGP, используемом для сопоставления меток MPLS с маршрутами для L3VPN и помеченных одноадресной передачей

Если chunks опция указана, вывод показывает текущий список фрагментов меток, предоставленных BGP Zebra, с указанием начальной и конечной метки в каждом фрагменте

Если inuse опция указана, на выходе отображается текущий список используемых сопоставлений меток с префиксами

Если ledger опция указана, вывод показывает список всех запросов ярлыков, выполненных для каждого префикса

Если requestsопция указана, на выходе отображается текущий список запросов меток, которые еще не были выполнены labelpool

Если summaryуказан параметр, вывод представляет собой сводку подсчетов для блоков, неиспользования, реестра и списка запросов, а также количество невыполненных запросов блоков к Zebra и количество повторных подключений zebra, которые произошли

Если jsonопция указана, выходные данные отображаются в формате JSON.

1.8.4.5.2 Отображение маршрутов по атрибуту большого сообщества

Следующие команды позволяют отображать маршруты на основе их атрибута большого сообщества.

show [ip] bgp <ipv4|ipv6> large-community

show [ip] bgp <ipv4|ipv6> large-community LARGE-COMMUNITY

show [ip] bgp <ipv4|ipv6> large-community LARGE-COMMUNITY

show [ip] bgp <ipv4|ipv6> large-community LARGE-COMMUNITY

Эти команды отображают маршруты BGP, которые имеют атрибут large community. атрибут. Когда `LARGE-COMMUNITY` указан, отображаются маршруты BGP, соответствующие этому большому сообществу. Когда *точное соответствие* указан, он отображает только маршруты, которые имеют точное совпадение. Когда *json* указан, он отображает маршруты в формате json.

show [ip] bgp <ipv4|ipv6> large-community-list

show [ip] bgp <ipv4|ipv6> large-community-list

show [ip] bgp <ipv4|ipv6> large-community-list

Эти команды отображают маршруты BGP для указанного семейства адресов, которые соответствуют указанному большому списку сообществ. Если указано точное совпадение, отображаются только те маршруты, которые имеют точное совпадение. Когда указан json, он отображает маршруты в формате json.

1.8.4.5.3 Отображение маршрутов в виде пути

show bgp ipv4|ipv6

Эта команда отображает маршруты BGP, соответствующие регулярному выражению *строка* (Регулярные выражения BGP).

show [ip] bgp ipv4 vpn

show [ip] bgp ipv6 vpn

Распечатайте активные маршруты IPV4 или IPV6, объявленные через VPN SAFI.

show bgp ipv4 vpn

show bgp ipv6 vpn

Распечатайте сводку соседних подключений для указанной комбинации AFI/ SAFI.

1.8.4.5.4 Отображение маршрутов с помощью средства различения маршрутов

show bgp [<ipv4|ipv6> vpn | l2vpn evpn [route]] rd <all|RD>

Для семейств адресов L3VPN и EVPN маршруты могут отображаться для каждого RD (Route Distinguisher) или для всех RD.

show bgp l2vpn evpn rd <all|RD> [overlay | tags]

Используйте `overlaytags` или ключевые слова для отображения информации о наложении / теге о префиксах EVPN в выбранном различителе маршрута.

show bgp l2vpn evpn routerd <all|RD> mac <MAC> [ip <MAC>] [json]

Для маршрутов EVPN типа 2 (macip) команде может быть предоставлен MAC-адрес (и, необязательно, IP-адрес), чтобы отображать только совпадающие префиксы в указанном RD.

1.8.4.5.5 Отображение информации о группе обновления

show bgp update-groups[advertise-queue|advertised-routes|packet-queue]

Отображение информации о каждой отдельной используемой группе обновлений. Если указан SUBGROUP-ID, отображается только информация об этой конкретной группе. Если указана рекламная очередь, отображается список маршрутов, которые необходимо отправить одноранговым узлам в группе обновлений, объявленные маршруты означают список маршрутов, которые мы отправили одноранговым узлам в группе обновлений, а packet-queue указывает список пакетов в очереди для отправки.

show bgp update-groups

Отображение информации о событиях группы обновлений в FRR.

1.8.4.5.6 Отображение информации о Nexthop

show [ip] bgp [<view|vrf> VIEWVRFNAME] nexthop ipv4 [A.B.C.D] [detail] [json]

show [ip] bgp [<view|vrf> VIEWVRFNAME] nexthop ipv6 [X:X::X:X] [detail] [json]

show [ip] bgp [<view|vrf> VIEWVRFNAME] nexthop [<A.B.C.D|X:X::X:X>] [detail] [json]

show [ip] bgp <view|vrf> all nexthop [json]

Отображение информации о nexthops для соседей bgp. Если указан определенный nexthop, также предоставляется информация о путях, связанных с nexthop. Опция с подробным описанием предоставляет информацию о воротах каждого nexthop.

show [ip] bgp [<view|vrf> VIEWVRFNAME] import-check-table [detail] [json]

Отображение информации о nexthops из таблицы, которая используется для проверки существования сети в операторах rib для сетевых операторов.

1.8.4.5.7 Сегмент-маршрутизация IPv6

show bgp segment-routing srv6

Эта команда отображает информацию о SRv6 L3VPN в bgpd. В частности, какой тип локатора используется, и информация о его фрагменте локатора. И SID функции SRv6, которая

фактически управляется в bgpd. In the following example, bgpd is using a Locator named loc1, and two SRv6 Functions are managed to perform VPNv6 VRF redirect for vrf10 and vrf20.

```
router# show bgp segment-routing srv6
locator_name: loc1
locator_chunks:
- 2001:db8:1:1::/64
functions:
- sid: 2001:db8:1:1::100
  locator: loc1
- sid: 2001:db8:1:1::200
  locator: loc1
bgps:
- name: default
  vpn_policy[AFI_IP].tovpn_sid: none
  vpn_policy[AFI_IP6].tovpn_sid: none
- name: vrf10
  vpn_policy[AFI_IP].tovpn_sid: none
  vpn_policy[AFI_IP6].tovpn_sid: 2001:db8:1:1::100
- name: vrf20
  vpn_policy[AFI_IP].tovpn_sid: none
  vpn_policy[AFI_IP6].tovpn_sid: 2001:db8:1:1::200
```

1.8.4.6 **Отображатель маршрута**

Маршрутизаторы BGP, подключенные внутри так же, как и через BGP, принадлежат внутреннему сеансу BGP или IBGP. Чтобы предотвратить циклы таблицы маршрутизации, IBGP не объявляет маршруты, изученные IBGP, другим маршрутизаторам в том же сеансе. Таким образом, IBGP требует полной сетки всех одноранговых узлов. Для больших сетей это быстро становится невозможным. Внедрение маршрутных отражателей устраняет необходимость в полной сетке.

Когда настроены отражатели маршрутов, они будут отражать маршруты, объявленные одноранговыми узлами, настроенными как клиенты. Клиент route reflector настроен с:

neighbor PEER route-reflector-client

Чтобы избежать одиночных точек отказа, можно сконфигурировать несколько отражателей маршрута.

Кластер представляет собой набор отражателей маршрутов и их клиентов и используется отражателями маршрутов, чтобы избежать зацикливания.

bgp cluster-id A.B.C.D

bgp no-rib

Чтобы установить и отключить демон -n/ параметры BGP --no_kernelво время выполнения, чтобы отключить установку маршрута BGP в RIB (Zebra), [no] bgp no-ribможно использовать команды;

Пожалуйста, обратите внимание, что установка параметра во время выполнения приведет к удалению всех маршрутов в ребре демонов из Zebra, а отключение его приведет к объявлению всех маршрутов в РЕБРЕ демонов в Zebra. Если параметр передается в качестве аргумента командной строки при запуске демона и конфигурация сохраняется, параметр будет сохранен, если он не будет удален из конфигурации с помощью отрицающей команды перед операцией записи конфигурации. На данный момент данные BGP, отличные от SAFI_UNICAST, не извлекаются должным образом из zebra при выполнении этой команды.

bgp allow-martian-nexthop

Когда одноранговый узел получает марсианский nexthop как часть NLRI для маршрута, разрешите использовать nextop как таковой, вместо отклонения и сброса соединения.

bgp send-extra-data zebra

Эта команда включает способность BGP отправлять дополнительные данные в zebra. В настоящее время это AS-Path, сообщества и причина выбора пути. Поведение по умолчанию в BGP - не отправлять эти данные. Если маршруты были отправлены в zebra и параметр изменен, bgpd не переустанавливает маршруты в соответствии с новой настройкой.

bgp session-dscp (0-63)

Эта команда позволяет bgp управлять на глобальном уровне значениями TCP dscp в заголовке TCP.

1.8.4.7 Подавление маршрутов, не установленных в FIB

Реализация FRR BGP передает префиксы, полученные от однорангового узла, другим одноранговым узлам, даже если маршруты не установлены в FIB. Могут быть сценарии, в которых аппаратные таблицы в некоторых маршрутизаторах (по пути от источника к месту назначения) заполнены, что приведет к тому, что все маршруты не будут установлены в FIB. Если эти маршруты будут объявлены нижестоящим маршрутизаторам, тогда трафик начнет поступать и будет отброшен на промежуточный маршрутизатор.

Решение заключается в предоставлении настраиваемой опции для проверки состояния установки префиксов FIB и объявления одноранговым узлам, если префиксы успешно установлены в FIB. Реклама префиксов подавляется, если она не установлена в FIB.

При проверке статуса установки маршрута в FIB будут применяться следующие условия:

Объявление или подавление маршрутов на основе статуса установки FIB применяется только к недавно изученным маршрутам от однорангового узла (маршрутам, которые не находятся в локальном RIB BGP).

Если маршрут, полученный от однорангового узла, уже существует в локальном RIB BGP и атрибуты маршрута изменились (изменен наилучший путь), старый путь удаляется, а новый путь устанавливается в FIB. Статус установки FIB не будет иметь никакого эффекта. Поэтому проверки применяются только при первом получении маршрута.

Эта функция не будет применяться к маршрутам, изученным с помощью других средств, таких как перераспределение в bgp из других протоколов. Это применимо только к одноранговым маршрутам.

Если маршрут установлен в FIB, а затем удаляется из плана данных, маршруты не будут удалены из одноранговых узлов. Это будет рассматриваться как проблема с плоскостью данных.

Эта функция немного увеличит время, необходимое для объявления маршрутов одноранговым узлам, поскольку статус установки маршрута должен быть получен из FIB

Если одноранговый узел получает маршруты до применения конфигурации, то сеансы bgp необходимо сбросить, чтобы конфигурация вступила в силу.

Если маршрут, который уже установлен в dataplane, по какой-либо причине удален, отправка сообщения об отзыве одноранговым узлам в настоящее время не поддерживается.

bgp suppress-fib-pending

Эта команда применима на глобальном уровне и на индивидуальном уровне bgp. При применении на глобальном уровне все экземпляры bgp будут ожидать установки fib перед объявлением маршрутов, и нет никакого способа отключить его для конкретного bgp vrf.

1.8.4.8 Политика маршрутизации

Вы можете установить другую политику маршрутизации для однорангового узла. Например, вы можете установить другой фильтр для однорангового узла.

```
!
! bgp 1 view 1
 neighbor  10.0.0.1 remote-as 2
```

```
 address-family ipv4 unicast
 neighbor  10.0.0.1 distribute-list 1 in
 exit-address-family
!
router bgp 1 view 2
 neighbor  10.0.0.1 remote-as 2
   ipv4 unicast
 neighbor 10.0.0.1 10.0.0.122exit-address-family
```

Это означает, что обновление BGP с одноранговой узла 10.0.0.1 распространяется как на BGP view 1, так и на view 2. Когда обновление вставляется в представление 1, применяется список распространения 1. С другой стороны, когда обновление вставляется в представление 2, применяется список распространения 2.

1.8.4.9 **Регулярные выражения BGP**

Регулярные выражения BGP основаны на POSIX 1003.2. Следующее описание - это всего лишь краткое подмножество регулярных выражений POSIX.

`.*`

Соответствует любому одиночному символу.

`*`

Соответствует 0 или более вхождениям шаблона.

`+`

Соответствует 1 или более вхождениям шаблона.

`?`

Сопоставьте 0 или 1 вхождений шаблона.

`^`

Соответствует началу строки.

`$`

Соответствует концу строки.

`_`

`_` Символ имеет особые значения в регулярных выражениях BGP. Он соответствует пробелу и запятой, а ТАКЖЕ в КАЧЕСТВЕ разделителя набора `{` и `}` и В КАЧЕСТВЕ разделителя конфедерации `(` и `)`. И он также соответствует началу строки и концу строки. So `_` может использоваться для СООТВЕТСТВИЯ границ значений. Этот символ технически оценивается `(^|[,{}()]|$)` как .

Различные примеры конфигурации

Пример сеанса для восходящего потока, рекламирующий только один префикс к нему.

```
router bgp 64512
 bgp router-id 10.236.87.1
   upstream peer-group
 neighbor upstream remote-as 64515
 neighbor upstream capability dynamic
```

```
 neighbor  10.1.1.1
 peer-group upstream
 neighbor  10.1.1.1 description ACME ISP

 address-family ipv4 unicast
 network 10.236.87.0/24
  neighbor upstream prefix-list pl-allowed-
adv out
 exit-address-family
!
ip prefix-list pl-allowed-adv seq 5 permit
82.195.133.0/25
ip prefix-list pl-allowed-adv seq 10 deny any
```

Более сложный пример, включающий сеансы восходящего потока, однорангового узла и клиента , рекламирующий глобальные префиксы и префиксы NO_EXPORT и предоставляющий действия для маршрутов клиентов на основе ценностей сообщества. Для поддержки выборочной рекламы префиксов широко используются карты маршрутов и функция ‘вызов’. Этот пример предназначен только для руководства, он НЕ был протестирован и почти наверняка содержит глупые ошибки, если не серьезные недостатки.

```
router bgp 64512
 bgp router-id 10.236.87.1
 neighbor upstream capability dynamic
 neighbor cust capability dynamic
 neighbor peer capability dynamic
 neighbor 10.1.1.1 remote-as 64515
 neighbor 10.1.1.1 peer-group upstream
 neighbor 10.2.1.1 remote-as 64516
 neighbor 10.2.1.1 peer-group upstream
 neighbor 10.3.1.1 remote-as 64517
 neighbor 10.3.1.1 peer-group cust-default
 neighbor 10.3.1.1 description customer1
 neighbor 10.4.1.1 remote-as 64518
 neighbor 10.4.1.1 peer-group cust
 neighbor 10.4.1.1 description customer2
 neighbor 10.5.1.1 remote-as 64519
 neighbor 10.5.1.1 peer-group peer
 neighbor 10.5.1.1 description peer AS 1
 neighbor 10.6.1.1 remote-as 64520
 neighbor 10.6.1.1 peer-group peer
 neighbor 10.6.1.1 description peer AS 2

 address-family ipv4 unicast
  network 10.123.456.0/24
  network 10.123.456.128/25 route-map rm-no-export
  neighbor upstream route-map rm-upstream-out out
  neighbor cust route-map rm-cust-in in
  neighbor cust route-map rm-cust-out out
  neighbor cust send-community both
  neighbor peer route-map rm-peer-in in
  neighbor peer route-map rm-peer-out out
  neighbor peer send-community both
  neighbor 10.3.1.1 prefix-list pl-cust1-network in
  neighbor 10.4.1.1 prefix-list pl-cust2-network in
  neighbor 10.5.1.1 prefix-list pl-peer1-network in
  neighbor 10.6.1.1 prefix-list pl-peer2-network in
 exit-address-family
!
ip prefix-list pl-default permit 0.0.0.0/0
!
ip prefix-list pl-upstream-peers permit 10.1.1.1/32
ip prefix-list pl-upstream-peers permit 10.2.1.1/32
!
ip prefix-list pl-cust1-network permit 10.3.1.0/24
ip prefix-list pl-cust1-network permit 10.3.2.0/24
!
ip prefix-list pl-cust2-network permit 10.4.1.0/24
```

```
!
ip prefix-list pl-peer1-network permit 10.5.1.0/24
ip prefix-list pl-peer1-network permit 10.5.2.0/24
ip prefix-list pl-peer1-network permit 192.168.0.0/24
!
ip prefix-list pl-peer2-network permit 10.6.1.0/24
ip prefix-list pl-peer2-network permit 10.6.2.0/24
ip prefix-list pl-peer2-network permit 192.168.1.0/24
ip prefix-list pl-peer2-network permit 192.168.2.0/24
ip prefix-list pl-peer2-network permit 172.16.1/24
!
bgp as-path access-list seq 5 asp-own-as permit ^$
bgp as-path access-list seq 10 asp-own-as permit _64512_
!
! ###################################################################
! Match communities we provide actions for, on routes receives from
! customers. Communities values of <our-ASN>:X, with X, have actions:
!
! 100 - blackhole the prefix
! 200 - set no_export
! 300 - advertise only to other customers
! 400 - advertise only to upstreams
! 500 - set no_export when advertising to upstreams
! 2X00 - set local_preference to X00
!
! blackhole the prefix of the route
bgp community-list standard cm-blackhole permit 64512:100
!
! set no-export community before advertising
bgp community-list standard cm-set-no-export permit 64512:200
!
! advertise only to other customers
bgp community-list standard cm-cust-only permit 64512:300
!
! advertise only to upstreams
bgp community-list standard cm-upstream-only permit 64512:400
!
! advertise to upstreams with no-export
bgp community-list standard cm-upstream-noexport permit 64512:500
!
! set local-pref to least significant 3 digits of the community
bgp community-list standard cm-prefmod-100 permit 64512:2100
bgp community-list standard cm-prefmod-200 permit 64512:2200
bgp community-list standard cm-prefmod-300 permit 64512:2300
bgp community-list standard cm-prefmod-400 permit 64512:2400
bgp community-list expanded cme-prefmod-range permit 64512:2...
!
! Informational communities
!
! 3000 - learned from upstream
! 3100 - learned from customer
! 3200 - learned from peer
!
bgp community-list standard cm-learnt-upstream permit 64512:3000
bgp community-list standard cm-learnt-cust permit 64512:3100
bgp community-list standard cm-learnt-peer permit 64512:3200
!
! ###################################################################
! Utility route-maps
!
! These utility route-maps generally should not used to permit/deny
! routes, i.e. they do not have meaning as filters, and hence probably
! should be used with 'on-match next'. These all finish with an empty
! permit entry so as not interfere with processing in the caller.
!
route-map rm-no-export permit 10
 set community additive no-export
route-map rm-no-export permit 20
!
```

```
route-map rm-blackhole permit 10
 description blackhole, up-pref and ensure it cannot escape this AS
 set ip next-hop 127.0.0.1
 set local-preference 10
 set community additive no-export
route-map rm-blackhole permit 20
!
! Set local-pref as requested
route-map rm-prefmod permit 10
 match community cm-prefmod-100
 set local-preference 100
route-map rm-prefmod permit 20
 match community cm-prefmod-200
 set local-preference 200
route-map rm-prefmod permit 30
 match community cm-prefmod-300
 set local-preference 300
route-map rm-prefmod permit 40
 match community cm-prefmod-400
 set local-preference 400
route-map rm-prefmod permit 50
!
! Community actions to take on receipt of route.
route-map rm-community-in permit 10
 description check for blackholing, no point continuing if it matches.
 match community cm-blackhole
 call rm-blackhole
route-map rm-community-in permit 20
 match community cm-set-no-export
 call rm-no-export
 on-match next
route-map rm-community-in permit 30
 match community cme-prefmod-range
 call rm-prefmod
route-map rm-community-in permit 40
!
! ################################################################
! Community actions to take when advertising a route.
! These are filtering route-maps,
!
! Deny customer routes to upstream with cust-only set.
route-map rm-community-filt-to-upstream deny 10
 match community cm-learnt-cust
 match community cm-cust-only
route-map rm-community-filt-to-upstream permit 20
!
! Deny customer routes to other customers with upstream-only set.
route-map rm-community-filt-to-cust deny 10
 match community cm-learnt-cust
 match community cm-upstream-only
route-map rm-community-filt-to-cust permit 20
!
! ################################################################
! The top-level route-maps applied to sessions. Further entries could
! be added obviously..
!
! Customers
route-map rm-cust-in permit 10
 call rm-community-in
 on-match next
route-map rm-cust-in permit 20
 set community additive 64512:3100
route-map rm-cust-in permit 30
!
route-map rm-cust-out permit 10
 call rm-community-filt-to-cust
 on-match next
route-map rm-cust-out permit 20
!
```

```
! Upstream transit ASes
route-map rm-upstream-out permit 10
 description filter customer prefixes which are marked cust-only
 call rm-community-filt-to-upstream
 on-match next
route-map rm-upstream-out permit 20
 description only customer routes are provided to upstreams/peers
 match community cm-learnt-cust
!
! Peer ASes
! outbound policy is same as for upstream
route-map rm-peer-out permit 10
 call rm-upstream-out
!
route-map rm-peer-in permit 10
 set community additive 64512:3200
```

Пример того, как настроить соединение с 6 костями.

```
! bgpd configuration
! ==================
!
! MP-BGP configuration
!
router bgp 7675
 bgp router-id 10.0.0.1
 neighbor 3ffe:1cfa:0:2:2a0:c9ff:fe9e:f56 remote-as `as-number`
!
 address-family ipv6
 network 3ffe:506::/32
 neighbor 3ffe:1cfa:0:2:2a0:c9ff:fe9e:f56 activate
 neighbor 3ffe:1cfa:0:2:2a0:c9ff:fe9e:f56 route-map set-nexthop out
 neighbor 3ffe:1cfa:0:2:2c0:4fff:fe68:a231 remote-as `as-number`
 neighbor 3ffe:1cfa:0:2:2c0:4fff:fe68:a231 route-map set-nexthop out
 exit-address-family
!
ipv6 access-list all permit any
!
! Set output nexthop address.
!
route-map set-nexthop permit 10
 match ipv6 address all
 set ipv6 nexthop global 3ffe:1cfa:0:2:2c0:4fff:fe68:a225
 set ipv6 nexthop local fe80::2c0:4fff:fe68:a225
!
log file bgpd.log
!
```

1.8.4.10 **Поддержка BGP tcp-mss**

Протокол TCP предоставляет пользователю механизм для указания максимального размера сегмента. setsockopt API используется для установки максимального размера сегмента для сеанса TCP. Мы можем настроить это как часть конфигурации соседей BGP.

В этом документе объясняется, как избежать проблем с уязвимостью ICMP путем ограничения максимального размера сегмента TCP при использовании обнаружения MTU. Использование обнаружения MTU в TCP-путях является одним из способов избежать фрагментации пакетов BGP.

Протокол TCP согласовывает значение максимального размера сегмента (MSS) во время установления сеансового соединения между двумя одноранговыми узлами. Согласованное значение MSS в первую очередь основано на максимальной единице передачи (MTU) интерфейсов, к которым напрямую подключены взаимодействующие одноранговые узлы. Однако из-за различий в MTU канала на пути, пройденном пакетами TCP, некоторые пакеты в сети, которые находятся в пределах значения MSS, могут быть фрагментированы, когда размер пакета превышает MTU канала.

Эта функция поддерживается с помощью TCP через IPv4 и TCP через IPv6.

1.8.4.10.1 Конфигурация командной строки

Приведенная ниже настройка может быть выполнена в режиме bgp маршрутизатора и позволяет пользователю настраивать значение tcp-mss для каждого соседа. Конфигурация применяется только после выполнения аппаратного сброса для этого соседа. Если мы настроим tcp-mss для обоих соседей, то оба соседа должны быть сброшены.

Конфигурация вступает в силу на основе приведенных ниже правил, поэтому для каждого сеанса TCP есть настроенное значение tcp-mss и синхронизированное значение tcp-mss.

По умолчанию, если настройка не выполнена, максимальный размер сегмента TCP устанавливается равным максимальной единице передачи (MTU) – (размер заголовка IP / IP6 + размер заголовка TCP + заголовок ethernet). Для IPv4 его MTU – (20 байт IP заголовок + 20 байт TCP заголовок + 12 байт ethernet заголовок), а для IPv6 его MTU – (40 байт IPv6 заголовок + 20 байт TCP заголовок + 12 байт ethernet заголовок).

Если настройка выполнена, она уменьшает 12-14 байт для заголовка ether и использует его после синхронизации в TCP handshake.

neighbor <A.B.C.D|X:X::X:X|WORD> tcp-mss (1-65535)

При настройке tcp-mss ядро уменьшает 12-14 байт для заголовка ethernet. Например, если tcp-mss настроен как 150, синхронизированное значение будет равно 138.

Примечание: настроенное и синхронизированное значение отличается, поскольку модуль TCP уменьшит 12 байт для заголовка ethernet.

1.8.4.10.2 запуск конфигурации

```
frr# show running-config
Building...

Current configuration:
!
router bgp 100
 bgp router-id 192.0.2.1
 neighbor  198.51.100.2 remote-as 100
 neighbor  198.51.100.2 tcp-mss 150 => new
entry
 neighbor  2001:DB8::2 remote-as 100
 neighbor  2001:DB8::2 tcp-mss 400 => new
entry
```

1.8.4.10.3 Показать команду

```
frr# show bgp neighbors 198.51.100.2
  neighbor is 198.51.100.2, remote AS 100, local AS 100, internal link
Hostname: frr
 BGP version 4, remote router ID 192.0.2.2, local router ID 192.0.2.1
  BGP state = Established, up for 02:15:28
  Last read 00:00:28, Last write 00:00:28
  Hold 180, keepalive interval is 60 seconds
 Configured tcp-mss is 150, synced tcp-mss is 138 => new display
```

```
frr# show bgp neighbors 2001:DB8::2
BGP - 2001:DB8::2, remote AS 100, local AS 100, internal link
Hostname: frr
 BGP version 4, remote router ID 192.0.2.2, local router ID 192.0.2.1
  BGP state = Established, up for 02:16:34
  Last read 00:00:34, Last write 00:00:34
  Hold 180, keepalive interval is 60 seconds
 Configured tcp-mss is 400, synced tcp-mss is 388 => new display
```

1.8.4.10.4 Показать вывод команды в формате json

```
frr# show bgp neighbors 2001:DB8::2 json
{
 "2001:DB8::2":{
 "remoteAs": 100, "localAs":100,
"nbrInternalLink":true,
 "hostname":"frr",
 "bgpVersion": 4,
"remoteRouterId": "192.0.2.2", "localRouterId": "192.0.2.1",
"bgpState":"Established",
 "bgpTimerUpMsec":8349000, "bgpTimerUpString":"02:19:09",
"bgpTimerUpEstablishedEpoch": 1613054251, "bgpTimerLastRead": 9000, "bgpTimerLastWrite":9000,
"bgpInUpdateElapsedTimeMsecs": 8347000, "bgpTimerHoldTimeMsecs":180000,
"bgpTimerKeepAliveIntervalMsecs":60000, "bgpTcpMssConfigured":400, => new entry
 "bgpTcpMssSynced":388, => new entry
```

```
frr# показать соседей bgp 198.51.100.2 в формате json
{
 "198.51.100.2":{
 "remoteAs": 100,
"localAs":100,
"nbrInternalLink": true,
"имя хоста": "frr",
"bgpVersion": 4,
"remoteRouterId": "192.0.2.2",
"localRouterId": "192.0.2.1",
"bgpState": "Установлено",
"bgpTimerUpMsec":8370000,
"bgpTimerUpString":"02:19:30",
" bgpTimerUpEstablishedEpoch": 1613054251,
"bgpTimerLastRead": 30000,
"bgpTimerLastWrite": 30000,
"bgpInUpdateElapsedTimeMsecs": 8368000,
"bgpTimerHoldTimeMsecs":180000,
"bgpTimerKeepAliveIntervalMsecs":60000,
"bgpTcpMssConfigured":150,                                  => new entry
    "bgpTcpMssSynced":138,                                  => new entry
```

1.8.4.11 Настройка FRR в качестве сервера маршрутизации

Целью сервера маршрутизации является централизация пиринга между динамиками BGP. Например, если у нас есть сценарий точки обмена с четырьмя динамиками BGP, каждый из которых поддерживает пиринг BGP с тремя другими (полная сетка), мы можем преобразовать его в централизованный сценарий, в котором каждый из четырех устанавливает единый пиринг BGP для сервера маршрутизации (сервер маршрутизации и клиенты).

Сначала мы кратко опишем модель сервера маршрутизации, реализованную FRR. Мы объясним команды, которые были добавлены для настройки этой модели. И, наконец, мы покажем полный пример FRR, настроенный как сервер маршрутизации.

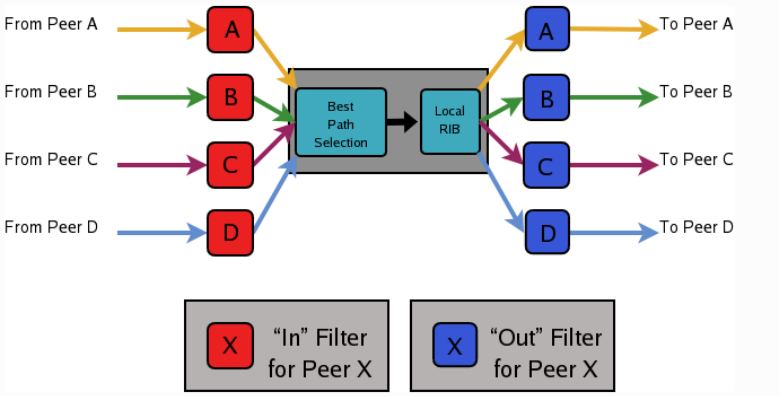
1.8.4.11.1 Описание модели сервера маршрутизации

Сначала мы собираемся описать обычную обработку, которой подвергаются объявления BGP внутри стандартного динамика BGP, как показано в разделе Обработка объявлений внутри "обычного" динамика BGP, она состоит из трех этапов:

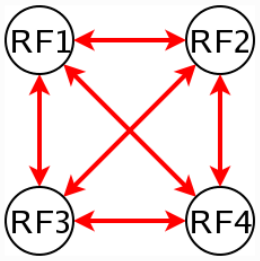
Когда объявление получено от некоторого одноранговового узла, к объявлению применяются фильтры In, настроенные для этого одноранговового узла. Эти фильтры могут отклонять объявление, принимать его без изменений или принимать его с некоторыми измененными атрибутами.

Объявления, прошедшие фильтры In, попадают в процесс выбора наилучшего пути, где они сравниваются с другими объявлениями, относящимися к тому же адресату, которые были получены от разных одноранговых узлов (в случае, если такие другие объявления существуют). Для каждого отдельного пункта назначения объявление, выбранное как лучшее, вставляется в Loc-RIB динамика BGP.

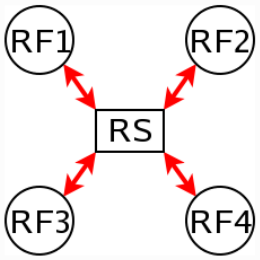
Маршруты, которые вставлены в Loc-RIB, рассматриваются для объявления всем одноранговым узлам (кроме того, от которого пришел маршрут). Это делается путем прохождения маршрутов в Loc-RIB через фильтры Out, соответствующие каждому узлу. Эти фильтры могут отклонять маршрут, принимать его без изменений или принимать его с некоторыми измененными атрибутами. Те маршруты, которые принимаются фильтрами выхода одноранговго узла, объявляются этому одноранговому узлу.



Обработка объявлений внутри 'обычного' динамика BGP¶



Полная сетка¶



Маршрутизировать сервер и клиентов¶

Рисунок 17

Конечно, мы хотим, чтобы таблицы маршрутизации, полученные на каждом из маршрутизаторов, были одинаковыми при использовании сервера маршрутизации, чем при его отсутствии. Но в результате наличия одного пиринга BGP (против сервера маршрутизации), динамики BGP больше не могут различать, от / к какому одноранговому узлу приходит / уходит каждое объявление.

Это означает, что маршрутизаторы, подключенные к серверу маршрутизации, не могут сами применять те же фильтры ввода / вывода, что и в сценарии с полной сеткой, поэтому они должны делегировать эти функции серверу маршрутизации.

Более того, выбор ‘наилучшего пути’ также должен выполняться внутри сервера маршрутизации от имени его клиентов. Причина в том, что если после применения фильтров диктора и (потенциального) получателя сервер маршрутизации решит отправить какому-либо клиенту два или более разных объявления, относящихся к одному и тому же адресату, клиент сохранит только последнее, рассматривая его как неявное удаление предыдущих объявлений длятот же пункт назначения. Это ожидаемое поведение динамика BGP, как определено вRFC 1771, и хотя есть некоторые предложения механизмов, которые позволяют отправлять несколько путей для одного и того же адресата через один BGP peering, ни один из них в настоящее время не поддерживается большинством существующих реализаций BGP.

Как следствие, сервер маршрутизации должен поддерживать дополнительную информацию и выполнять дополнительные задачи для RS-клиента, помимо тех, которые необходимы для обычных BGP peerings. По сути, сервер маршрутизации должен:

Поддерживайте отдельную информационную базу маршрутизации (Loc-RIB) для каждого одноранговового узла, настроенного как RS-клиент, содержащую маршруты, выбранные в результате процесса "Выбора наилучшего пути", который выполняется от имени этого RS-клиента.

Всякий раз, когда он получает сообщение от RS-клиента, он должен учитывать его для локальных сетей других RS-клиентов.

Это означает, что для каждого из них сервер маршрутизации должен пропустить объявление через соответствующий выходной фильтр диктора.

Затем через соответствующий фильтр потенциального получателя.

Только если объявление будет принято обоими фильтрами, оно будет передано в процесс выбора наилучшего пути.

Наконец, это может попасть в Loc-RIB приемника.

Когда мы говорим о ‘соответствующем’ фильтре, необходимо учитывать как диктора, так и получателя маршрута. Предположим, что сервер маршрутизации получает объявление от клиента A, и сервер маршрутизации рассматривает его для локального ребра клиента B. Фильтры, которые должны быть применены, являются теми же, которые использовались бы в сценарии с полной сеткой, т. Е. Сначала Выходной фильтр маршрутизатора A для объявлений, поступающих на маршрутизатор B, а затем входной фильтр маршрутизатора B для объявлений, поступающих с маршрутизатора A.

Мы вызываем "Политику экспорта" RS-клиента в набор фильтров вывода, которые клиент использовал бы, если бы не было сервера маршрутизации. То же самое относится к ‘Политике импорта’ RS-клиента и набору встроенных фильтров клиента, если не было сервера маршрутизации.

Также часто от сервера маршрутизации требуется, чтобы он не изменял некоторые атрибуты BGP (next-hop, as-path и MED), которые обычно изменяются стандартными динамиками BGP перед объявлением маршрута.

Модель обработки объявлений, реализованная FRR, показана в модели обработки объявлений, реализованной сервером маршрутизации. На рисунке показана смесь RS-клиентов (B, C и D) с обычными одноранговыми узлами BGP (A). Есть некоторые детали, которые заслуживают дополнительных комментариев:

Объявления, поступающие от обычного одноранговового узла BGP, также рассматриваются для локальных сетей всех RS-клиентов. Но логически они не проходят через какую-либо политику экспорта.

Те одноранговые узлы, которые настроены как RS-клиенты, не получают никаких сообщений от основного Loc-RIB.

Помимо политик импорта и экспорта, для RS-клиентов также могут быть установлены входные и выходные фильтры. Фильтры In могут быть полезны, когда сервер маршрутизации

также имеет обычные одноранговые узлы BGP. С другой стороны, фильтры Out для RS-клиентов, вероятно, не нужны, но мы решили не удалять их, поскольку они никому не вредят (их всегда можно оставить пустыми).

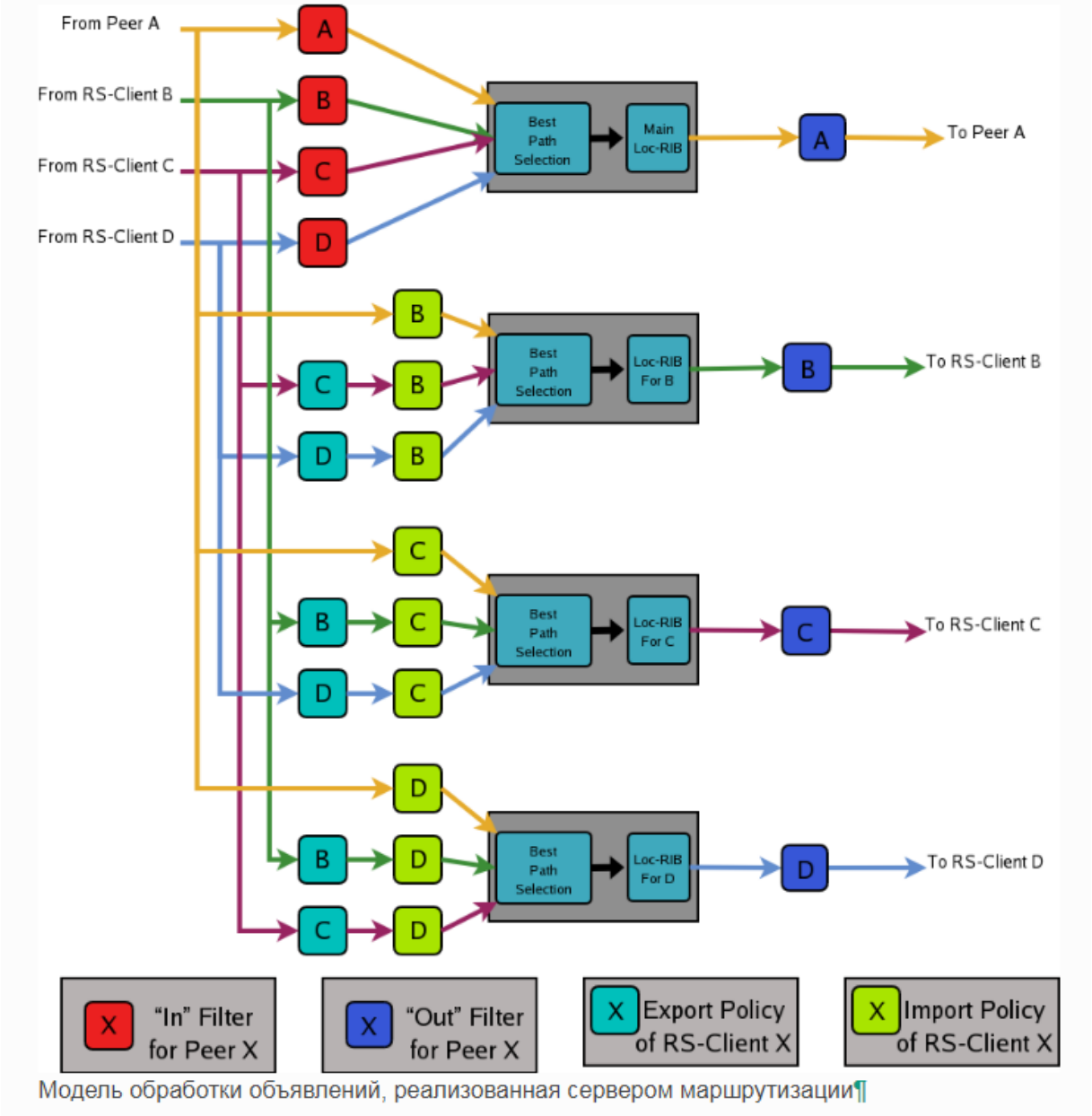


Рисунок 18

1.8.4.11.2 Команды для настройки сервера маршрутизации

Теперь мы опишем команды, которые были добавлены в frr для поддержки функций сервера маршрутизации.

neighbor PEER-GROUP route-server-client

neighbor A.B.C.D route-server-client

neighbor X:X::X:X route-server-client

Эта команда настраивает одноранговый узел, заданный peer, A.B.C.D или X: X:: X:X в качестве RS-клиента.

На самом деле эта команда не новая, она уже существовала в стандартном FRR. Он включает прозрачный режим для указанного однорангового узла. Это означает, что некоторые атрибуты BGP (as-path, next-hop и MED) маршрутов, объявленных этому узлу, не изменяются.

С исправлением сервера маршрутизации эта команда, помимо установки прозрачного режима, создает новый Loc-RIB, выделенный для указанного однорангового узла (с именами Loc-RIB для X в Модель обработки объявлений, реализованная сервером маршрутизации.). Начиная с этого момента, каждое объявление, полученное сервером маршрутизации, также будет учитываться для нового Loc-RIB.

neigbor A.B.C.D|X.X::X.X|peer-group route-map

Этот набор команд можно использовать для указания карты маршрутов, представляющей политику импорта или экспорта однорангового узла, который настроен как RS-клиент (с помощью предыдущей команды).

match peer A.B.C.D|X:X::X:X

Это новый совпадение инструкция для использования в маршрутных картах, позволяющая им описывать политики импорта/экспорта. Как мы говорили ранее, политика импорта / экспорта представляет собой набор фильтров ввода / вывода RS-клиента. Это утверждение делает возможным, чтобы одна карта маршрута представляла полный набор фильтров, которые динамик BGP использовал бы для своих разных одноранговых узлов в сценарии, отличном от RS.

The сопоставление с одноранговым оператор имеет разную семантику, независимо от того, используется ли он внутри карты маршрутов импорта или экспорта. В первом случае оператор совпадает, если адрес однорангового узла, который отправляет сообщение, совпадает с адресом, указанным в {A.B.C.D|X:X::X:X} . Для карт маршрутов экспорта это соответствует, когда {A.B.C.D |X: X::X:X} является адресом RS-клиента, в чье Loc-ребро будет вставлено объявление (как одна и та же политика экспорта применяется до того, как в объявлении отображаются разные Loc-ребрамодель обработки, реализованная сервером маршрутизации.).

call

Эта команда (также используемая внутри карты маршрутов) переходит в другую карту маршрутов, имя которой задается *WORD*. Когда вызываемая карта маршрута завершается, в зависимости от ее результата, исходная карта маршрута продолжается или нет. Помимо того, что эта команда полезна для упрощения написания карт маршрутов импорта / экспорта, ее также можно использовать внутри любой обычной (входящей или исходящей) карты маршрутов.

1.8.4.11.3 Пример конфигурации сервера маршрутизации

Наконец, мы собираемся показать, как настроить демон FRR для работы в качестве сервера маршрутизации. Для этой цели мы представим сценарий без сервера маршрутизации, а затем покажем, как использовать конфигурации маршрутизаторов BGP для создания конфигурации сервера маршрутизации.

Все файлы конфигурации, показанные в этом разделе, взяты из сценариев, которые были протестированы с помощью инструмента VNUML http://www.dit.upm.es/vnuml ,ВНУМЛ.

1.8.4.11.4 Настройка маршрутизаторов BGP без сервера маршрутизации

Предположим, что наш первоначальный сценарий представляет собой точку обмена с тремя маршрутизаторами, поддерживающими BGP, с именами RA, RB и RC. Каждый из динамиков BGP генерирует некоторые маршруты (с помощью команды network) и устанавливает пиринговые соединения BGP с двумя другими маршрутизаторами. Для этих пирингов настроены входные и выходные карты маршрутов с именами типа ‘PEER-X-IN’ или ‘PEER-X-OUT’. Например, файл конфигурации для маршрутизатора RA может быть следующим:

```
#Configuration for router 'RA'

!
hostname RA
password ****
!
router bgp 65001
  no bgp default ipv4-unicast
 neighbor  2001:0DB8::B remote-as 65002
  neighbor  2001:0DB8::C remote-as 65003
!
  address-family ipv6
 network  2001:0DB8:AAAA:1::/64
 network  2001:0DB8:AAAA:2::/64
 network  2001:0DB8:0000:1::/64
 network  2001:0DB8:0000:2::/64
 neighbor  2001:0DB8::B activate
 neighbor  2001:0DB8::B soft-reconfiguration inbound
 neighbor  2001:0DB8::B route-map PEER-B-IN in
 neighbor  2001:0DB8::B route-map PEER-B-OUT out
  neighbor2001
:0DB8::C soft-reconfiguration inbound
 neighbor  2001:0DB8::C route-map PEER-C-IN in
 neighbor 2001:0DB8::C route-map PEER-C-OUT out
 exit-address-family
!
ipv6 prefix-list COMMON-PREFIXES seq 5 permit 2001:0DB8:0000::/48 ge 64 le 64
ipv6 prefix-list COMMON-PREFIXES seq 10 deny any

!
ipv6 prefix-list PEER-A-PREFIXES seq 5 permit 2001:0DB8:AAAA::/48 ge 64 le 64
ipv6 prefix-list PEER-A-PREFIXES seq 10 deny any
!
ipv6645642001:0DB8:BBBB::/48 ge 10
ipv6 prefix-list PEER-B-PREFIXES seq 10
!!
20015 permit 2001: le 64
ipv66410 deny any
10
route-map
!10
  match10set metric
route-map10020
  match20set community
65001:11111 !
route-map
!10
  10 ipv6 address prefix-list COMMON-PREFIXES
 set metric 200
route-map2020
  match ipv6 address prefix-list PEER-C-PREFIXES
 set community 65001:22222

!
route-map 10
  match!!
route-map 10
  match ipv6 address prefix-list PEER-A-PREFIXES
!
line vty

!
```

1.8.4.11.5 Настройка маршрутизаторов BGP с сервером маршрутизации

Чтобы преобразовать первоначальный сценарий в сценарий с сервером маршрутизации, сначала мы должны изменить конфигурацию маршрутизаторов RA, RB и RC. Теперь они не

должны обмениваться данными между собой, а только с сервером маршрутизации. Например, конфигурация RA превратится в:

```
# Configuration for router 'RA'
!
hostname RA password ****
!
router bgp 65001
  no bgp default ipv4-unicast  neighbor  2001:0DB8::FFFF remote-as 65000
!
  address-family ipv6  network  2001:0DB8:AAAA:1::/64  network  2001:0DB8:AAAA:2::/64  network
2001:0DB8:0000:1::/64  network  2001:0DB8:0000:2::/64  neighbor  2001:0DB8::FFFF activate
neighbor  2001:0DB8::FFFF soft-reconfiguration inbound  exit-address-family
!
line vty

!
```

Что логически намного проще, чем его первоначальная конфигурация, поскольку теперь он поддерживает только один пиринг BGP, а все фильтры (карты маршрутов) исчезли.

1.8.4.11.6 Конфигурация самого сервера маршрутизации

Как мы уже говорили, когда описывали функции сервера маршрутизации (Описание модели сервера маршрутизации), он отвечает за всю фильтрацию маршрутов. Для достижения этого входные и выходные фильтры из конфигураций RA, RB и RC должны быть преобразованы в политики импорта и экспорта на сервере маршрутизации.

Это фрагмент конфигурации сервера маршрутизации (мы показываем только политики для RA клиента):

```
# Configuration for Route Server ('RS')

!
hostname RS
password ix
!
router bgp 65000 view RS
 no bgp default ipv4-unicast
 neighbor  2001:0DB8::A remote-as 65001
  neighbor  2001:0DB8::B remote-as 65002
  neighbor  2001:0DB8::C remote-as 65003
!
  address-family ipv6
 neighbor  2001:0DB8::A activate
  neighbor2001 2001 2001:0DB8::A route-map RSCLIENT-A-IMPORT in
 neighbor  2001:0DB8::A route-map RSCLIENT-A-EXPORT out
 neighbor  2001:0DB8::A soft-reconfiguration inbound

 neighbor  2001:0DB8::B activate
 neighbor  2001:0DB8::B route-server-client
 neighbor  2001:0DB8::B route-map RSCLIENT-B-IMPORT in
 neighbor  2001:0DB8::B route-map RSCLIENT-B-EXPORT out
 neighbor  2001:0DB8::B soft-reconfiguration inbound

 neighbor  2001:0DB8::C activate
 neighbor  2001:0DB8::C route-server-client
 neighbor  2001:0DB8::C route-map RSCLIENT-C-IMPORT in
 neighbor  2001:0DB8::C route-map RSCLIENT-C-EXPORT out
 neighbor  2001:0DB8::C soft-reconfiguration inbound
 exit-address-family
!
ipv6 prefix-list COMMON-PREFIXES seq 5 permit 2001:0DB8:0000::/48 ge 64 le 64
ipv6 prefix-list COMMON-PREFIXES seq 10 deny any
!
ipv6 prefix-list PEER-A-PREFIXES seq 5 permit 2001:0DB8:AAAA::/48 ge 64 le 64
ipv6 prefix-list PEER-A-PREFIXES seq 10 deny any
```

```
!
ipv6 prefix-list PEER-B-PREFIXES seq 5 permit 2001:0DB8:BBBB::/48 ge 64 le 64
ipv6 prefix-list PEER-B-PREFIXES seq 10 deny any
!
ipv6 prefix-list PEER-C-PREFIXES seq 5 permit 2001:0DB8:CCCC::/48 ge 64 le 64
ipv6 prefix-list PEER-C-PREFIXES seq 10 deny any
!
route-map RSCLIENT-A-IMPORT permit 10
  match peer 2001:0DB8::B
 call A-IMPORT-FROM-B
route-map RSCLIENT-A-IMPORT permit 20
  match peer 2001:0DB8::C
 call A-IMPORT-FROM-C
!
route-map A-IMPORT-FROM-B permit 10
  match ipv6 address prefix-list COMMON-PREFIXES
 set metric 100
route-map A-IMPORT-FROM-B permit 20
  match20set community
:11111
65001
:11111
!10
  match10set metric ,
route-map200
  20match20set community 65001:22222
65001
:22222
!10
  10 peer 2001:0DB8::B
 match ipv6 address prefix-list PEER-A-PREFIXES
route-map RSCLIENT-A-EXPORT permit 20
  match peer 2001:0DB8::C
 match ipv6 address prefix-list PEER-A-PREFIXES
!
...
...
...
```

Если вы сравните начальную конфигурацию RA с конфигурацией сервера маршрутизации, приведенной выше, вы увидите, насколько легко создать импорт и Экспорт политик для RA из маршрутов входа и выхода -карты исходной конфигурации RA.

Когда не было сервера маршрутизации, RA поддерживал два пиринга, один с RB, а другой с RC. Для каждого из этих пирингов была настроена карта маршрута. Чтобы создать карту маршрута импорта для RA клиента на сервере маршрутов, просто добавьте записи карты маршрутов, следуя этой схеме:

```
route-map <NAME> permit 10
    match peer <Peer Address>
    call <In Route-Map for this Peer>
route-map <NAME> permit 20
    match peer <Another Peer Address>
    call <In Route-Map for this Peer>
```

Это именно тот процесс, который был выполнен для создания route-map RSCLIENT-A-IMPORT. Карты маршрутов, которые вызываются внутри него (A-IMPORT-FROM-B и A-IMPORT-FROM-C), точно такие же, как в маршрутных картах из исходной конфигурации RA (PEER-B-IN и PEER-C-IN), только имяотличается.

То же самое можно было бы сделать для создания политики экспорта для RA (route-map RSCLIENT-A-EXPORT), но в этом случае исходные карты маршрутов были настолько простыми, что мы решили не использовать команды call WORD, и мы объединили все в единую карту маршрутов (RSCLIENT-A-ЭКСПОРТ).

Политики импорта и экспорта для RB и RC не показаны, но процесс будет идентичным.

1.8.4.11.7 Дополнительные соображения о картах маршрутов импорта и экспорта

Текущая версия исправления сервера маршрутов позволяет указывать карту маршрутов только для политик импорта и экспорта, в то время как в стандартном BGP-динамике помимо карт маршрутов есть другие инструменты для выполнения фильтрации ввода и вывода (списки доступа, списки сообществ, ...). Но это не представляет никаких ограничений, поскольку все виды фильтров могут быть включены в карты маршрутов импорта / экспорта. Например, предположим, что в сценарии без маршрутизирующего сервера одноранговый узел RA имеет следующие фильтры, настроенные для ввода от одноранговго узла B:

```
2001 2001:0DB8::B prefix-list LIST-1 in
neighbor  2001:0DB8::B filter-list LIST-2 in
neighbor  2001:0DB8::B route-map PEER-B-IN in
...
...
route-map 10
  match ipv6 address prefix-list COMMON-PREFIXES
   local-preference 100
route-map 20
  match ipv6 address prefix-list PEER-B-PREFIXES
 set community 65001:11111
```

Можно написать единственную карту маршрутов, которая эквивалентна трем фильтрам (списку сообщества, списку префиксов и карте маршрутов). Затем эту карту маршрутов можно использовать внутри политики импорта на сервере маршрутизации. Давайте посмотрим, как это сделать:

```
2001 2001:0DB8::A route-map RSCLIENT-A-IMPORT in
...
!
...
route-map RSCLIENT-A-IMPORT permit 10
  match peer 2001:0DB8::B
 call A-IMPORT-FROM-B
...
...
!
route-map A-IMPORT-FROM-B permit 1
  match ipv6 address prefix-list LIST-1
 match as-path LIST-2
 on-match1010
2 A-IMPORT-FROM-B deny
2route-map A-IMPORT-FROM-B permit 10
  match ipv6 address prefix-list COMMON-PREFIXES
 set local-preference 100
route-map20
  match ipv6 address prefix-list PEER-B-PREFIXES
 set community 65001: 11111
!
...
...
```

Карта маршрута A-IMPORT-FROM-B эквивалентна трем фильтрам (LIST-1, LIST-2 и PEER-B-IN). Первая запись карты маршрута A-IMPORT-FROM-B (порядковый номер 1) совпадает тогда и только тогда, когда совпадают СПИСОК префиксных списков-1 и список фильтров-2. Если это произойдет, из-за инструкции ‘on-match goto 10’ следующая запись карты маршрута, подлежащая обработке, будет иметь номер 10, и с этого момента карта маршрута A-IMPORT-FROM-B идентична PEER-B. Если первая запись не соответствует, on-match goto 10’ будет проигнорирован, и следующей обработанной записью будет номер 2, который запретит маршрут.

Таким образом, результат тот же, что и с тремя исходными фильтрами, т.Е., Если либо СПИСОК-1, либо СПИСОК-2 отклоняет маршрут, он не достигает одноранговго узла карты

маршрута. В случае, если и LIST-1, и LIST-2 принимают маршрут, он передается одноранговому узлу, который может отклонить, принять или изменить маршрут.

1.8.4.12 **Проверка происхождения префикса с использованием RPKI**

Проверка источника префикса позволяет маршрутизаторам BGP проверять, является ли источник С IP-префикса законным для объявления этого IP-префикса. Требуемые объекты аттестации хранятся в инфраструктуре открытых ключей ресурсов (RPKI). Однако маршрутизаторы с поддержкой RPKI не хранят сами криптографические данные, а только информацию о проверке. Проверка криптографических данных (так называемая авторизация источника маршрута, или короткая ROA, объекты) будут выполняться доверенными серверами кэша. Протокол RPKI/ RTR определяет стандартный механизм для поддержания обмена префиксом / источником В ВИДЕ сопоставления между сервером кэша и маршрутизаторами. В сочетании со схемой проверки происхождения префикса BGP маршрутизатор может проверять полученные обновления BGP, не страдая от криптографической сложности.

Протокол RPKI / RTR определен в RFC 6810 и схема проверки в RFC 6811. Текущая версия проверки происхождения префикса в FRR реализует оба RFC.

Для получения более подробной, но все же простой для чтения справочной информации мы предлагаем:

[Обеспечение безопасности-BGP]

[Сертификация ресурсов]

1.8.4.12.1 **Особенности текущей реализации**

В двух словах, текущая реализация предоставляет следующие возможности

Маршрутизатор BGP может подключаться к одному или нескольким серверам кэша RPKI для получения проверенного префикса к источнику в КАЧЕСТВЕ сопоставлений. Расширенный переход на другой ресурс может быть реализован серверными сокетами с различными значениями предпочтений.

Если по истечении заданного времени ожидания не удается установить соединение с сервером кэша RPKI, маршрутизатор будет обрабатывать маршруты без проверки источника префикса. Он по-прежнему будет пытаться установить соединение с сервером кэша RPKI в фоновом режиме.

По умолчанию включение RPKI не изменяет выбор наилучшего пути. В частности, недопустимые префиксы по-прежнему будут учитываться при выборе наилучшего пути. Однако маршрутизатор можно настроить так, чтобы он игнорировал все недопустимые префиксы.

Карты маршрутов могут быть настроены в соответствии с определенным состоянием проверки RPKI. Это позволяет создавать локальные политики, которые обрабатывают маршруты BGP на основе результатов проверки источника префикса.

Обновления с серверов кэша RPKI применяются напрямую, и выбор пути обновляется соответствующим образом. (Мягкая реконфигурация необходимо должна быть включена, чтобы это работало).

1.8.4.12.2 **Включение RPKI**

Вы должны установить frr-rpki-rtrlibдополнительный пакет для поддержки RPKI, иначе bgpddaemon не запустится.

rpki

Эта команда включает режим настройки RPKI. Большинство команд, начинающихся с rpki может использоваться только в этом режиме.

Когда он используется в сеансе telnet, выход из этого режима приводит к инициализации rpki.

Выполнение этой команды само по себе не активирует проверку префикса. Необходимо настроить хотя бы один доступный сервер кэша. См. Раздел Настройка серверов кэширования RPKI/RTR.

Не забудьте добавить -M rpki к переменной bgpd_optionsв /etc/frr/daemons, например, так:

bgpd_options=" -A 127.0.0.1 -M rpki"

вместо настройки по умолчанию:

bgpd_options=" -A 127.0.0.1"

В противном случае вы столкнетесь с ошибкой при попытке войти в режим конфигурации RPKI из-за rpkitого, что модуль не загружается при инициализации демона BGP.

Примеры ошибки:

```
router (config)# debug rpki
% [BGP] Unknown command: debug rpki

router (config)# rpki
% [BGP] Unknown command: rpki
```

Обратите внимание, что команды RPKI будут доступны во vtysh при запуске `find rpki` независимо от того, загружен ли модуль.

1.8.4.12.3 Настройка серверов кэширования RPKI/RTR

Следующие команды не зависят от конкретного сервера кэша.

rpki polling_period (1-3600)

Установите количество секунд, в течение которых маршрутизатор ожидает, пока маршрутизатор снова не запросит у кэша обновленные данные.

Значение по умолчанию равно 300 секундам.

rpki expire_interval (600-172800)

Установите количество секунд, в течение которых маршрутизатор ожидает, пока у маршрутизатора не истечет срок действия кэша.

Значение по умолчанию равно 7200 секундам.

rpki (1-7200)

Установите количество секунд, в течение которых маршрутизатор ожидает повторной попытки подключения к серверу кэша.

Значение по умолчанию равно 600 секундам.

rpki cache(A.B.C.D|WORD) PORT [SSH_USERNAME] [SSH_PRIVKEY_PATH] [KNOWN_H

OSTS_PATH] [source A.B.C.D] preference (1-255)

Добавьте сервер кэша в сокет. По умолчанию соединение между маршрутизатором и сервером кэша основано на обычном TCP. Защита соединения между маршрутизатором и сервером кэша с помощью SSH не является обязательной. Удаление сокета удаляет связанный сервер кэша и прерывает существующее соединение.

A.B.C.D|WORD

Адрес сервера кэширования.

ПОРТ

Номер порта для подключения к серверу кэша

SSH_USERNAME

Имя пользователя SSH для установления SSH-соединения с сервером кэша.

SSH_PRIVKEY_PATH

Локальный путь, который включает в себя файл закрытого ключа маршрутизатора.

KNOWN_HOSTS_PATH

Локальный путь, включающий известный файл hosts. Значение по умолчанию зависит от конфигурации среды операционной системы, обычно `~/.ssh/known_hosts`.

источник A.B.C.D

Исходный адрес подключения RPKI к серверу кэша доступа.

1.8.4.12.4 Проверка обновлений BGP

match rpki notfound|invalid|valid

Создайте предложение для карты маршрута, чтобы сопоставить префиксы с указанным состоянием RPKI.

В следующем примере маршрутизатор предпочитает допустимые маршруты недопустимым префиксам, поскольку недопустимые маршруты имеют более низкие локальные предпочтения.

```
! Allow
!  for invalid routes in route selection processroute bgp 60001
 !
! Set local preference of invalid prefixes to 10
route-map 10
 match rpki invalid  set local-preference 10
 !
! Set local preference of valid prefixes to 500
route-map 500
 match rpki valid
 set local-preference 500
```

match rpki-extcommunity notfound|invalid|valid

Создайте предложение для карты маршрута, чтобы сопоставить префиксы с указанным состоянием RPKI, которое является производным от атрибута расширенного сообщества исходного состояния проверки (OVS). Расширенное сообщество OVS не является транзитивным и обменивается только между узлами iBGP.

1.8.4.12.5 Отладка

debug rpki

Включить или отключить отладочный вывод для RPKI.

1.8.4.12.6 Отображение RPKI

show rpki prefix<A.B.C.D/M|X:X::X:X/M> [(1-4294967295)] [json]

Отображение проверенных префиксов, полученных с серверов кэша, отфильтрованных по указанному префиксу.

show rpki as-number ASN [json]

Отображение проверенных префиксов, полученных с серверов кэша, отфильтрованных по ASN.

show rpki prefix-table[json]

Отображать все проверенные префиксы к источнику В ВИДЕ сопоставлений / записей, которые были получены с серверов кэша и сохранены в маршрутизаторе. На основе этих данных маршрутизатор проверяет обновления BGP.

show rpki cache-server[json]

Отображение всех настроенных серверов кэша, независимо от того, активны они или нет.

show rpki cache-connection[json]

Отобразите все подключения к кэшу и покажите, какое из них подключено, а какое нет.

show bgp [afi] [safi] <A.B.C.D|A.B.C.D/M|X:X::X:X|X:X::X:X/M> rpki <valid|invalid|notfound>

Отображение для указанного префикса или адреса путей bgp, которые соответствуют заданному состоянию rpki.

show bgp [afi] [safi] rpki <valid|invalid|notfound>

Отобразить все префиксы, соответствующие заданному состоянию rpki.

1.8.4.12.7 Пример конфигурации RPKI

```
hostname bgpd1 password zebra ! log stdout
debug bgp updates debug bgp keepalives debug rpki !
rpki
 rpki polling_period 1000
 rpki10
 ! SSH Example:
  rpki cache example.com source 141.22.28.223  22 rtr-ssh ./ssh_key/id_rsa ./ssh_key/id_rsa.pub
preference 1
 ! TCP Example:
  rpki cache rpki-validator.realmv6.org 8282 preference 2
  exit
!
router bgp 60001
 bgp router-id 141.22.28.223
 network  192.168.0.0 / 16
 neighbor  123.123.123.0 remote-as 60002
 neighbor 123.123.123.0 route-map rpki in  neighbor  123.123.123.0 update-source 141.22.28.223
!
 address-family ipv6  neighbor  123.123.123.0 activate  neighbor  123.123.123.0 route-map rpki in
 exit-address-family
!
route-map10
 match rpki invalid  set local-preference 10
!
route-map20
 match rpki notfound  set local-preference 20
!
route-map 30
 match rpki valid  set local-preference 30
!
route-map40
!
```

1.8.4.12.8 Обзор

При обычном многолучевом распространении с равной стоимостью (ECMP) маршрут к месту назначения имеет несколько следующих переходов, и ожидается, что трафик будет равномерно распределен по этим следующим переходам. На практике хеширование на основе потоков используется для того, чтобы весь трафик, связанный с определенным потоком, использовал один и тот же следующий переход и, соответственно, один и тот же путь по сети.

Взвешенный протокол ECMP с использованием пропускной способности канала BGP обеспечивает поддержку многолучевого распространения по всей сети с неравной стоимостью (UCMP) к IP-адресату. Неравномерное распределение нагрузки по стоимости реализуется плоскостью пересылки на основе весов, связанных со следующими переходами IP-префикса. Эти веса вычисляются на основе полос пропускания соответствующих многолучевых каналов, которые кодируются в BGP link bandwidth extended communitysоответствии с [Проект-IETF-idr-link-bandwidth]. Замена соответствующего значения пропускной способности канала BGP на префикс в сети приводит к неравномерной стоимости многолучевого распространения в масштабах всей сети.

Один из основных вариантов использования этой возможности - в центре обработки данных, когда служба (представленная своим IP-адресом anycast) имеет неодинаковый набор ресурсов в регионах (например, POD) центра обработки данных, и сама сеть обеспечивает функцию балансировки нагрузки вместо внешнего балансировщика нагрузки. Обратитесь к [Draft-IETF-mohanty-bess-ebgp-dmz] и RFC 7938 для получения подробной информации об этом варианте использования. Этот вариант использования применим как в чистой сети L3, так и в сети EVPN.

Также поддерживается традиционный вариант использования пропускной способности канала BGP для балансировки нагрузки трафика на выходные маршрутизаторы в AS на основе пропускной способности их внешних одноранговых каналов eBGP.

1.8.4.12.9 Принципы проектирования

Вычисление и использование веса следующего перехода

Как описано, в UCMP существует вес, связанный с каждым следующим переходом IP-префикса, и ожидается, что трафик будет распределен по следующим переходам пропорционально их весу. Вес следующего перехода - это простое разложение полосы пропускания соответствующего канала на общую полосу пропускания всех многолучевых каналов, отображенную в диапазоне от 1 до 100. Что произойдет, если не все пути в многолучевом наборе имеют связанную с ними пропускную способность канала? В таком случае, в соответствии с[Проект-IETF-idr-link-bandwidth], поведение возвращается к стандартному ECMP среди всех многолучевых каналов, при этом пропускная способность канала фактически игнорируется.

Обратите внимание, что нет никаких изменений ни в алгоритме выбора наилучшего пути BGP, ни в алгоритме вычисления многолучевости; сопоставление пропускной способности канала с весом происходит во время установки маршрута в RIB.

Если пересылка данных реализована с помощью ядра Linux, при вычислении хэша используется вес следующего перехода. Ядро использует алгоритм порога хэширования, и в него встроено использование веса следующего перехода; следующие переходы не нужно расширять для достижения UCMP. UCMP для IPv4 также доступен в более старых ядрах Linux, в то время как UCMP для IPv6 доступен начиная с ядра 4.16.

Если пересылка данных реализована аппаратно, обычные реализации расширяют следующие переходы (т. е. они повторяются) в контейнере ECMP пропорционально их весу. Например, если веса, связанные с 3 следующими переходами для определенного маршрута, равны 50, 25 и 25, а контейнер ECMP имеет размер 16 следующих переходов, первый

следующий переход будет повторен 8 раз, а остальные 2 следующих перехода повторяются по 4 раза каждый. Также возможны другие реализации.

Неравномерная стоимость многолучевого распространения по сети

Для перечисленных выше вариантов использования недостаточно поддерживать UCMP только на одном маршрутизаторе (например, выходном маршрутизаторе) или по отдельности на нескольких маршрутизаторах; UCMP должен быть развернут по всей сети. Это достигается за счет использования сообщества BGP link-bandwidth extended.

На маршрутизаторе, который генерирует пропускную способность канала BGP, должна быть пользовательская конфигурация для его запуска, которая описана ниже. Принимающие маршрутизаторы будут использовать полученную полосу пропускания канала от своих нижестоящих маршрутизаторов для определения веса следующего перехода, как описано в предыдущем разделе. Кроме того, если полученная пропускная способность канала является транзитивным атрибутом, она будет передана одноранговым узлам eBGP с дополнительным изменением, заключающимся в том что если следующий переход установлен для себя, совокупная пропускная способность канала всех нижестоящих путей передается другим маршрутизаторам. Таким образом, вся сеть будет знать, как распределять трафик для службы anycast по сети.

Сообщество с расширенной пропускной способностью канала BGP кодируется в байтах в секунду. В случае использования, когда UCMP должен основываться на количестве путей, используется эталонная пропускная способность 1 Мбит / с. Так, например, если существует 4 равных по стоимости пути к любому IP-адресу, кодированная полоса пропускания в расширенном сообществе составит 500 000. Само фактическое значение не имеет значения, если все маршрутизаторы, передающие пропускную способность канала, делают это одинаково.

1.8.4.12.10 Руководство по настройке

Настройка взвешенного ECMP с использованием пропускной способности канала BGP требует одного важного шага - использования карты маршрутов для внедрения расширенного сообщества с пропускной способностью канала. Предусмотрена дополнительная опция для управления обработкой полученной полосы пропускания канала.

Ввод пропускной способности канала в сеть

На маршрутизаторе “точки входа”, который вводит префикс, к которому должна выполняться взвешенная балансировка нагрузки, должна быть настроена карта маршрутов для подключения сообщества с расширенной пропускной способностью канала.

Для варианта использования обеспечения взвешенной балансировки нагрузки для службы anycast эту конфигурацию обычно необходимо применять на маршрутизаторе TOR или Leaf, который подключен к серверам, предоставляющим службу anycast, а пропускная способность будет зависеть от количества многолучевых каналов для назначения.

Для варианта использования балансировки нагрузки на выходной маршрутизатор, выходной маршрутизатор должен быть настроен с картой маршрутов, указывающей значение полосы пропускания a, которое соответствует пропускной способности канала, соединяющегося с его узлом eBGP в соседнем AS. Кроме того, сообщество с расширенной пропускной способностью канала должно быть явно настроено как нетранзитивное.

Полный синтаксис команды route-map set можно найти в расширенных сообществах BGP в Route Map

Эта карта маршрутов поддерживается только в двух точках подключения: (a) карта исходящих маршрутов, подключенная к одноранговому узлу или группе одноранговых узлов, для каждого семейства адресов (b) карта маршрутов EVPN advertise, используемая для введения одноадресных маршрутов IPv4 или IPv6 в EVPN в качестве маршрутов типа 5.

Поскольку создание полосы пропускания канала осуществляется с помощью карты маршрутов, она может быть ограничена определенными префиксами (например, только для служб anycast) или может быть сгенерирована для всех префиксов. Кроме того, когда карта маршрутов используется в контексте соседей, использование полосы пропускания канала может быть ограничено только определенными узлами.

Пример конфигурации показан ниже и иллюстрирует объявление полосы пропускания канала для одноранговой группы “SPINE” для любых IP-адресов в диапазоне 192.168.x.x

```
ip prefix-list anycast_ip seq 10 permit 192.168.0.0 / 16 le 32
route-map anycast_ip permit 10
 match ip address prefix-list anycast_ip
 set extcommunity bandwidth num-multipaths
route-map anycast_ip permit 20
!
router bgp 65001
 neighbor SPINE peer-group
 neighbor SPINE remote-as external
  neighbor172.16.35.1 172.16.35.1 172.16.36.1 peer-group SPINE
!
 address-family ipv4 unicast
 network 110.0.0.1/32


!
```

Управление обработкой полосы пропускания канала на приемнике

Нет конфигурации, необходимой для обработки полученной полосы пропускания канала и преобразования ее в вес, связанный с соответствующим следующим переходом; это происходит по умолчанию. Если некоторые из многолучевых каналов не имеют расширенной пропускной способности канала, поведение по умолчанию заключается в возврате к обычному ECMP, как рекомендовано в [Проект-IETF-idr-link-bandwidth].

Оператор может изменить это поведение с помощью следующей конфигурации:

bgp bestpath bandwidth <ignore | skip-missing | default-weight-for-missing>

Различные варианты подразумевают следующее поведение:

игнорировать: полностью игнорировать пропускную способность канала для установки маршрута (т. Е. Выполнять обычный ECMP, не взвешенный)

пропуск-отсутствует: пропустить пути без пропускной способности канала и выполнить UCMP среди других (если хотя бы некоторые пути имеют пропускную способность канала)

значение по умолчанию для отсутствующих: назначьте низкий вес по умолчанию (значение 1) путям, не имеющим пропускной способности канала

Эта конфигурация для каждого экземпляра BGP аналогична другим элементам управления выбором маршрута BGP; в этом экземпляре она работает как с одноадресными IPv4, так и с одноадресными IPv6 маршрутами. В сети EVPN эта конфигурация (если требуется) должна быть реализована в VRF клиента и снова применима для одноадресной рассылки IPv4 и одноадресной рассылки IPv6, включая те, которые получены из маршрутов EVPN типа 5.

Примерный фрагмент конфигурации FRR на приемнике для пропуска путей без пропускной способности канала и выполнения взвешенного ECMP среди других путей (если некоторые из них имеют пропускную способность канала), как показано ниже.

```
router bgp 65021
 bgp bestpath as-path multipath-relax
 bgp bestpath bandwidth skip-missing
 neighbor LEAF peer-group
 neighbor LEAF remote-as external
 neighbor 172.16.35.2 peer-group LEAF
 neighbor 172.16.36.2 peer-group LEAF
 !
 address-family ipv4 unicast
  network 130.0.0.1/32
```

```
 exit-address-family
!
```

Остановка распространения полосы пропускания канала за пределы домена

Сообщество с расширенной пропускной способностью канала будет автоматически распространяться с префиксом на одноранговые узлы EBGP, если создатель закодировал его как транзитивный атрибут. Если это распространение должно быть остановлено за пределами определенного домена (например, прекращено распространение на маршрутизаторы за пределами базовой сети центра обработки данных), доступный механизм заключается в отключении рекламы всех расширенных сообществ BGP на определенных одноранговых сетях. Другими словами, распространение не может быть заблокировано только для сообщества с расширенной пропускной способностью канала. Конфигурация для отключения всех расширенных сообществ может быть применена к одноранговому узлу или одноранговой группе (для каждого семейства адресов).

Конечно, другой распространенный способ остановить распространение полосы пропускания канала за пределы домена - это заблокировать рекламу самих префиксов и, возможно, объявить только совокупный маршрут. Это было бы довольно распространенным явлением в сети EVPN.

Мониторинг и устранение неполадок пропускной способности канала BGP и UCMP

Существующие операционные команды для отображения таблицы маршрутизации BGP для определенного префикса также будут отображать сообщество с расширенной пропускной способностью канала, если оно присутствует.

Ниже показан пример одноадресного маршрута IPv4, полученного с атрибутом пропускной способности канала от двух одноранговых узлов:

```
CLI# show bgp ipv4 unicast 192.168.10.1/32
BGP routing table entry for 192.168.10.1/32
Paths: (2 available, best #2, table default)
 Advertised to non peer-group peers:
 l1(swp1) l2(swp2) l3(swp3) l4(swp4)
 65002
    fe80::202:ff:fe00:1b from l2(swp2) (110.0.0.2)
 (fe80::202:ff:fe00:1b) (used)
 Origin IGP, metric 0, valid, external, multipath, bestpath-from-AS 65002
      Extended Community: LB:65002:125000000 (1000.000 Mbps)
 Last update: Thu Feb 20 18:34:16  2020

   65001
    fe80::202:ff:fe00:15 from l1(swp1) (110.0.0.1)
 (fe80::202:ff:fe00:15) (used)
 Origin IGP, metric 0, valid, external, multipath, bestpath-from-AS 65001, best (Older Path)
 Extended Community: LB:65001:62500000 (500.000 Mbps)
 Last update: Thu Feb 20 18:22:34  2020
```

Веса, связанные со следующими переходами маршрута, можно увидеть, запросив у RIB конкретный маршрут.

Например, веса следующего перехода, соответствующие ширине полосы пропускания канала в приведенном выше примере, показаны ниже:

```
spine1# show ip route 192.168.10.1/32
Routing entry for 192.168.10.1/32
  Known via "bgp", distance 20, metric 0, best
  Last update 00:00:32 ago
  * fe80::202:ff:fe00:1b, via swp2, weight 66
  * fe80::202:ff:fe00:15, via swp1, weight 33
```

Для устранения неполадок можно использовать существующие журналы отладки `debug bgp updates`, `debug bgp bestpath <prefix>`, `debug bgp zebra` и `debug zebra kernel`

Ниже показан фрагмент журнала отладки, когда он `debug bgp zebra` включен, и маршрут устанавливается BGP в RIB с весами следующего перехода:

```
:26:190
2020-02-29T060.928096+00:00 leaf1 bgpd[5459]: Tx route add VRF :26:19.928096 192.168.150.1/3233
192.168.150.1/32 tag 0 count 0
2020-02-29T06:26:19.928289.928289+00:00 leaf1 bgpd[5459]: nhop [1]: 110,0 if 35 VRF 33 wt 50 RMAC
0a:11:2f:7d:35:20
2020-02-29T06:26:19.928479.928479+00:00 leaf1 bgpd[5459]: nhop [2]: 110.0.0.5 if 35 VRF 33 wt 50
RMAC 32:1e:32:a3:6c:bf
2020-02-29T06:26:19.928668.928668+00:00 leaf1 bgpd[5459]: bgp_zebra_announce: 192.168.150.1 /32:
announcing to zebra (recursion NOT set)
```

1.8.4.13 **Спецификация потока**

1.8.4.13.1 Обзор

Flowspec представляет новый формат кодирования NLRI, который используется для распространения спецификаций потока правил трафика. По сути, вместо того, чтобы просто полагаться на IP-адрес назначения для IP-префиксов, IP-префикс заменяется n-кортежем, состоящим из правила. Это правило может представлять собой более или менее сложную комбинацию следующих:

Сетевой источник / назначение (может быть одним или другим, или обоими).

Информация уровня 4 для UDP / TCP: порт источника, порт назначения или любой другой порт.

Информация уровня 4 для типа ICMP и кода ICMP.

Информация уровня 4 для флагов TCP.

Информация уровня 3: значение DSCP, тип протокола, длина пакета, фрагментация.

Разные флаги TCP уровня 4.

Обратите внимание, если изначально в Flowspec были определены правила IPv4, также возможно использовать семейство адресов IPv6. Можно использовать тот же набор комбинаций, что и для IPv4.

Для фильтрации трафика применяется комбинация вышеуказанных правил. Это кодируется как часть определенных расширенных сообществ BGP, и действия могут варьироваться от очевидного перенаправления (на nexthop или для разделения VRF) до формирования или отбрасывания.

Следующие проекты IETF и RFC были использованы для реализации FRR Flowspec:

RFC 5575

[Проект-IETF-IDR-Flowspec-перенаправление-IP]

[Проект-IETF-IDR-Flow-Spec-V6]

1.8.4.13.2 Принципы проектирования

FRR реализует клиентскую часть Flowspec, то есть BGP может получать записи Flowspec, но не может выступать в качестве менеджера и отправлять записи Flowspec.

Linux предоставляет следующие механизмы для реализации маршрутизации на основе политик:

Фильтрация трафика с Netfilterпомощью . Netfilterпредоставляет набор инструментов, таких как ipsetiptables, которые достаточно мощны, чтобы иметь возможность фильтровать такое правило фильтра Flowspec.

использование нестандартных таблиц маршрутизации с помощью iproute2(с помощью ip rulекоманды, предоставленной iproute2). iproute2уже используется демоном PBR FRR, который обеспечивает базовую маршрутизацию на основе политик на основе IP-источника и критерия назначения.

Приведенный ниже пример является иллюстрацией того, что Flowspec будет внедряться в базовую систему:

```
# linux shell
```

```
ipsetcreate  match0x102hash:net,net counters
ipset add match0x102 32.0.0.0 /16,40.0.0.0/16
iptables -N match0x102  -tmangle
iptables -A match0x102  -tmangle -j MARK --set-mark 102
iptables -A match0x102 - mangle -j ACCEPT
iptables -i ntfp3 -t mangle -I PREROUTING -m set --match-set match0x102
             src,dst -g match0x102
 iprule add fwmark 102 lookup 102

ip route add 40.0.0.0 /16 via 44.0.0.2 .0.0.2table 102
```

Для обработки входящей записи Flowspec применяется следующий рабочий процесс:

Входящие записи Flowspec обрабатываются bgpd, хранящаяся в BGP RIB.

Запись Flowspec устанавливается в соответствии с ее сложностью.

Он будет установлен, если в расширенном сообществе BGP будет замечено одно из следующих действий фильтрации: перенаправление IP или перенаправление VRF в сочетании с опцией rate для перенаправления трафика. Или параметр rate, установленный на 0, для отбрасывания трафика.

В зависимости от степени сложности записи Flowspec, она будет установлена в zebra РЕБРО. Для получения дополнительной информации о том, что поддерживается в реализации FRR как правило, см. Ограничения / Известные проблемы. Запись Flowspec разбивается на несколько частей перед отправкой в zebra.

демон zebra получает конфигурацию маршрутизации политики

Объекты маршрутизации на основе политик, необходимые для маршрутизации трафика в базовой системе, принимаются zebra. В : и context будут созданы или добавлены два контекста фильтрацииNetfilteripsetiptable. Первый используется для определения IP-фильтра на основе нескольких критериев. Например, набор IPnet:net-адресов основан на двух IP-адресах, в то время net,port,netкак он основан на двух IP-адресах и одном порту (для ICMP, UDP или TCP). Способ использования фильтрации (например, используется порт src или порт dst?) Определяется последним контекстом фильтрации. iptablекоманда будет ссылаться на ipsetконтекст и расскажет, как фильтровать и что делать. В нашем случае будет установлен маркер, указывающийiproute2, куда перенаправлять трафик. Иногда для удаления действия нет необходимости добавлять маркер; iptablebудет указано удалить все пакеты, соответствующие ipset записи.

1.8.4.13.3 Руководство по настройке

Для настройки механизма IPv4 Flowspec используйте следующую конфигурацию. На сегодняшний день возможно настроить Flowspec только на VRF по умолчанию.

```
router bgp <AS>
 neighbor <A.B.C.D> remote-as <remoteAS>
 neighbor <A:B::C:D> remote-as <remoteAS2>
 address-family ipv4 flowspec
 neighbor <A.B.C.D> activate
 exit
  address-family ipv6 flowspec
 neighbor <A:B::C:D> activate
 exit
exit
```

Вы можете просмотреть записи Flowspec, используя одну из следующих команд show:

show bgp ipv4 flowspec[detail | A.B.C.D]

show bgp ipv6 flowspec[detail | A:B::C:D]

1.8.4.13.3.1 Конфигурация для каждого интерфейса

Одной из приятных функций является возможность применения Flowspec к определенному интерфейсу, вместо того, чтобы применять его ко всей машине. Несмотря на следующий проект IETF [Проект-IETF-IDR-Flowspec-Interface-Set] не реализовано, можно вручную ограничить приложение Flowspec некоторыми входящими интерфейсами. На самом деле, неиспользование этого может привести к некоторому неожиданному поведению, например, к удвоению трафика или замедлению трафика (затраты на фильтрацию). Чтобы ограничить Flowspec одним конкретным интерфейсом, используйте следующую команду в разделе *семейство адресов flowspec* узел.

local-install <IFNAME | any>

По умолчанию Flowspec активирован на всех интерфейсах. Установка его в именованный интерфейс приведет к разрешению только этого интерфейса. И наоборот, включение любого интерфейса приведет к удалению всех ранее настроенных интерфейсов.

1.8.4.13.3.2 Перенаправление VRF

Еще одна приятная функция для настройки - это возможность перенаправлять трафик на отдельный VRF. Эта функция не противоречит возможности настройки Flowspec только для VRF по умолчанию. На самом деле, когда вы получаете входящие записи BGP flowspec в этом VRF по умолчанию, вы можете перенаправить трафик на другой VRF.

Напомним, что записи BGP flowspec имеют расширенное сообщество BGP, которое содержит цель маршрута. Поиск локального VRF на основе целевого маршрута заключается в следующем:

Необходимо выполнить настройку каждого VRF с установленным целевым назначением маршрута. Каждый VRF настраивается в экземпляре BGP VRF со своим собственным списком целевых маршрутов. Принятый целевой формат маршрута соответствует следующему: A.B.C.D:U16, или U16:U32, U32:U16.

Для маршрутизации трафика будет выбран первый VRF с соответствующей целью маршрута. Используйте следующую команду в разделе одноадресный адрес ipv4 -семейный узел

rt redirectimport RTLIST...

Чтобы проиллюстрировать, если целевой объект маршрута, настроенный в записи Flowspec, `E.F.G.H:II`, затем будет установлен экземпляр BGP VRF с той же целью маршрута. Затем будет выбран этот VRF. Приведенный ниже пример полной конфигурации показывает, как настраиваются целевые объекты маршрута и как выполняется настройка VRFS и перекрестная настройка VRF. Обратите внимание, что VRF отображаются в сетевых пространствах имен Linux. Чтобы трафик данных пересекал границы VRF, создаются виртуальные интерфейсы Ethernet с частной схемой IP-адресации.

```
router bgp <ASx>
 neighbor <A.B.C.D> remote-as <ASz>
 address-family ipv4 flowspec
 neighbor A.B.C.D activate
 exit

exitrouter bgp <ASy> vrf vrf2
 address-family ipv4 unicast
 rt redirect import <E.F.G.H:II>
 exit
exit
```

Аналогично, можно сделать то же самое для правил IPv6 flowspec, используя расширенное сообщество IPv6. Формат определен в **RFC 5701**, и это сообщество содержит адрес IPv6, закодированный в атрибуте, и соответствует локально настроенному целевому IPv6

импортированного маршрута, определенному в соответствующем Экземпляр BGP VRF. Приведенный ниже пример определяет расширенное сообщество IPv6, содержащее Адрес *E: F :: G: H*, за которым следуют 2 байта, выбранные администратором (здесь *JJ*).

```
router bgp <ASx>
 neighbor <A:B::C:D> remote-as
<ASz>
 address-family ipv6 flowspec
  neighbor A:B::C:D activate
 exit
exit
router bgp <ASy> vrf vrf2
 address-family ipv6 unicast
  rt6 redirect import <E:F::G:H:JJ>
 exit
exit
```

1.8.4.13.3.3 Мониторинг и устранение неполадок Flowspec

Вы можете отслеживать объекты маршрутизации политики с помощью одной из следующих команд. Эти команды полагаются на контексты фильтрации, настроенные из BGP, и получают статистическую информацию, полученную из базовой системы. Другими словами, эти статистические данные извлекаются из Netfilter.

show pbr ipsetIPSETNAME | iptable

`IPSETNAME` является ли имя объекта маршрутизации политики созданным `ipset`. Что касается контекстов правил, можно узнать, какое правило настроено для маршрутизации определенного трафика. `show pbr iptable` Команда отображает для пересылаемого трафика, какая таблица используется. Затем легко использовать этот идентификатор таблицы для дампа таблицы маршрутизации, которой будет соответствовать пересылаемый трафик.

show ip route tableTABLEID

`TABLEID` является идентификатором номера таблицы, ссылающимся на нестандартную таблицу маршрутизации, используемую в этом примере.

debug bgp flowspec

Вы можете устранить неполадки в Flowspec или маршрутизации на основе политики BGP. Например, если вы столкнулись с некоторыми проблемами при декодировании записи Flowspec, вы должны включить `debug bgp flowspec`.

debug bgp pbr [error]

Если вам не удастся применить запись flowspec в zebraдолжна быть какая-то связь с механизмом маршрутизации политики. Здесь, debug bgp pbr error может помочь.

Чтобы получить информацию о созданных/удаленных контекстах маршрутизации политики, используйте только debug bgp pbr команда.

Как показано ниже, можно проверить правильность установки записи Flowspec и правильность маршрутизации входящего трафика в соответствии с политикой. Прежде всего, вы должны проверить, установлена ли запись Flowspec или нет.

```
CLI# show bgp ipv4 flowspec 5.5.5.2 /32
 BGP flowspec entry: (flags 0x418)  Destination Address  5.5.5.2 / 32
   IP Protocol = 17
   Destination> = 50, <= 90
   FS:redirect VRF RT:255.255.255.255:255  received for 18:41:37
   installed in PBR (match0x271ce00)
```

Это означает, что запись Flowspec была установлена в `iptable` названный `match0x271ce00`. Получив подтверждение, что он установлен, вы можете

проверить, найдена ли соответствующая запись, выполнив следующую команду. Вы также можете проверить, был ли сопоставлен входящий трафик, посмотрев на строку счетчика.

```
CLI# show pbr ipset match0x271ce00
IPset match0x271ce00 type net,port
 to  5.5.5.0/24:proto 6: 80-120 (8) pkts  1000, bytes 1000000
     to  5.5.5.2:proto 17:50-90 (5)
pkts  1692918, bytes 157441374
```

Как вы можете видеть, запись присутствует. обратите внимание, что `iptable` запись может использоваться для размещения нескольких записей Flowspec. Чтобы узнать, куда перенаправляется соответствующий трафик, вы должны ознакомиться с правилами маршрутизации политики. Маршрутизация политики выполняется путем перенаправления трафика на номер таблицы маршрутизации. Этот номер таблицы маршрутизации достигается с помощью `iptable`. Связь между номером таблицы маршрутизации и входящим трафиком `MARKER` задается таблицей IP, ссылающейся на IPSet. В случае Flowspec `iptable` ссылки на `ipset` контекст имеют то же имя. Таким образом, легко узнать, какая таблица маршрутизации используется, выполнив следующую команду:

```
CLI# show pbr iptable
   IPtable match0x271ce00 action redirect (5)
     pkts 1700000, bytes 158000000
     table 257, fwmark 257
...
```

Как вы можете видеть, используя следующие команды Linux, МАРКЕР 0x101присутствует в обоих iptableip ruleконтекстах и .

```
# iptables -t mangle --list match0x271ce00 -v
Chain match0x271ce00 (1 references)
pkts bytes target     prot opt in     out     source               destination
1700K  158M MARK       all  --  any    any     anywhere             anywhere
     MARK set 0x101
1700K  158M ACCEPT     all  --  any    any     anywhere             anywhere

# ip rule list
0:from all lookup local
0:from all fwmark 0x101 lookup 257
32766:from all lookup main
32767:from all lookup default
```

Это позволяет нам видеть, куда перенаправляется трафик.

1.8.4.14 **Ограничения / Известные проблемы**

Как вы можете видеть, Flowspec богат и может быть очень сложным. На сегодняшний день не все правила Flowspec можно преобразовать в действия маршрутизации на основе политик.

NetfilterДрайвер еще не интегрирован в FRR. Отсутствие этого фрагмента кода предотвращает внедрение записей flowspec в базовую систему.

Существуют некоторые ограничения, связанные с фильтрацией контекстов

Если я беру пример портов UDP или портов TCP в Flowspec, информация может быть диапазоном портов или уникальным значением. Это дело рассмотрено. Однако сложность может быть увеличена, если поток представляет собой комбинацию списка диапазонов портов и перечисления уникальных значений. Здесь этот случай не обрабатывается. Аналогично, невозможно создать фильтр как для порта src, так и для порта dst. Например, отфильтруйте порт src из [1-1000] и порт dst = 80. Такая же сложность невозможна для длины пакета, типа ICMP, кода ICMP.

Есть некоторые другие известные проблемы:

Процедура проверки, описанная в RFC 5575, недоступна.

Эта процедура проверки не была реализована, поскольку эта функция не использовалась в существующих настройках, которыми вы поделились с нами.

Значение формирователя действия фильтрации, если оно положительное, не используется для применения формирования.

Если значение положительное, трафик перенаправляется в желаемое место назначения без каких-либо других действий, настроенных Flowspec. При необходимости рекомендуется настроить качество обслуживания более глобально для каждого интерфейса.

При неожиданном сбое или другом событии у zebra может не быть времени для очистки контекстов PBR.

То есть ipset, iptableи ip rule контексты. Это также является следствием того факта, что ip-правило / ipset / iptables не обнаруживаются при запуске (невозможно прочитать соответствующие контексты, поступающие из Flowspec).

1.8.4.15 **Приложение**

Дополнительная информация с публичной презентацией, которая объясняет дизайн Flowspec внутри FRRouting.

[Презентация]

Проект-IETF-IDR-Flowspec-перенаправление-IP
<https://tools.ietf.org/id/draft-ietf-idr-flowspec-redirect-ip-02.txt >

Проект-IETF-IDR-Flowspec-Interface-Set
<https://tools.ietf.org/id/draft-ietf-idr-flowspec-interfaceset-03.txt >

Проект-IETF-IDR-Flow-Spec-V6
<https://tools.ietf.org/id/draft-ietf-idr-flow-spec-v6-10.txt >

Презентация
<https://docs.google.com/presentation/d/1ekQygUAG5yvQ3wWUyrw4Wcag0LgmbW1kV02IWcU4iUg/edit#slide=id.g378f0e1b5e_1_44>

1
Для того, чтобы некоторый набор объектов имел порядок, должно существовать некоторое двоичное отношение упорядочения, которое определено для каждой комбинации этих объектов, и это отношение должно быть транзитивным. Т.е.: если оператор отношения равен < , и если a < b и b < c, то это отношение должно нестии должно быть, что a < c, чтобы объекты имели порядок. Отношение упорядочения может допускать равенство, т. Е. a < b и b < a могут быть истинными и подразумевать, что a и b равны по порядку и не различаются им, и в этом случае набор имеет частичный порядок. В противном случае, если есть порядок, все объекты имеют отдельное место в порядке, а набор имеет общий порядок)

bgp-route-osci-cond
Макферсон, Д. и Гилл, В. и Уолтон, Д., “Условие постоянных колебаний маршрута протокола пограничного шлюза (BGP)”, IETF RFC3345

стабильный-гибкий-ibgp
Флавель, А. и М. Рауэн, “Стабильный и гибкий iBGP”, ACM SIGCOMM 2009

правильность ibgp

Гриффин, Т. и Г. Уилфонг, “О правильности конфигурации IBGP”, ACM SIGCOMM 2002

1.8.4.16 **Поддержка быстрой конвергенции BGP**

Всякий раз, когда одноранговый адрес BGP становится недоступным, мы должны немедленно прервать сеанс BGP. В настоящее время немедленно отключаются только однократные сеансы EBGP.Сеансы IBGP и EBGP с несколькими переходами ожидают истечения таймера ожидания, чтобы завершить сеансы.

Этот новый параметр конфигурации помогает пользователю немедленно прерывать сеансы BGP, когда одноранговый узел становится недоступным.

bgp

Эта конфигурация доступна на уровне bgp. При включении конфигурация применяется ко всем соседям, настроенным в этом экземпляре bgp.

```
router bgp 64496
 neighbor  10.0.0.2 remote-as 64496
 neighbor fd00::2 remote-as 64496
 bgp fast-convergence
!
address-family ipv4 unicast  redistribute static exit-address-family

! neighborfd00::2 activate exit-address-family
```

1.8.5 **LDP**

Демон ldpd - это стандартизированный протокол, который позволяет обмениваться информацией о метках MPLS между устройствами MPLS. Протокол LDP обеспечивает пиринг между устройствами для обмена информацией о метках. Эта информация хранится в таблице MPLS zebra, и она вводит эту информацию MPLS в базовую систему (например, ядро Linux или систему OpenBSD). ldpd предоставляет необходимые опции для создания VPN уровня 2 в сети MPLS. Например, можно соединить несколько сайтов, которые используют один и тот же широковещательный домен.

FRR реализует LDP, как описано в RFC 5036; другими стандартами LDP являются следующие: RFC 6720, RFC 6667, RFC 5919, RFC 5561, RFC 7552, RFC 4447. Поскольку MPLS уже доступен, FRR также поддерживает RFC 3031.

1.8.5.1 **Запуск Ldpd**

Ldpd демон может быть вызван с помощью любой из распространенных опций (Общие параметры вызова).

--ctl_socket

Этот параметр позволяет переопределить путь к файлу ldpd.sock, используемому для управления этим демоном. Если этот параметр указан, он переопределяет добавление пути к параметру -N.

The zebra демон должен быть запущен до вызывается ldpd.

Настройка ldpd выполняется в его файле конфигурации ldpd.conf.

1.8.5.2 **Понимание принципов LDP**

Давайте сначала представим некоторые определения, которые позволят лучше понять протокол LDP:

LSR : Помеченный маршрутизатор коммутатора. Сетевые устройства, обрабатывающие метки, используемые для пересылки трафика между ними и через них.

*LER*Labeled Edge Router. A Labeled edge router is located at the edge of

Сеть MPLS, обычно между IP-сетью и сетью MPLS.

LDP предназначена для обмена информацией о ярлыках между устройствами. Он пытается установить пиринг с удаленными устройствами, поддерживающими LDP, сначала путем обнаружения с помощью UDP-порта 646, затем путем пиринга с использованием TCP-порта 646. Как только сеанс TCP установлен, информация о ярлыке передается через рекламные объявления ярлыков.

Существуют различные способы отправки режимов рекламы этикеток. Реализация фактически поддерживает следующее: либеральное сохранение ярлыков + нежелательный нисходящий поток + Независимый контроль. Другие рекламные режимы показаны ниже и сравниваются с текущей реализацией.

Либеральное сохранение ярлыков в сравнении с консервативным режимом В либеральном режиме каждая метка, отправленная каждым LSR, сохраняется в таблице MPLS. В консервативном режиме в таблице MPLS сохраняется только метка, которая была отправлена лучшим следующим переходом (определяется метрикой IGP) для этого конкретного FEC.

Независимое управление LSP в сравнении с упорядоченным управлением LSP MPLS имеет два способа привязки меток к FEC; либо через упорядоченное управление LSP, либо независимое управление LSP. Упорядоченный элемент управления LSP привязывает метку к FEC только в том случае, если это выходной LSR, или маршрутизатор получил привязку метки для FEC от маршрутизатора следующего перехода. В этом режиме маршрутизатор MPLS создаст привязку метки для каждого FEC и распространит ее среди своих соседей, если у него есть запись в RIB для назначения. В другом режиме привязки меток выполняются без каких-либо зависимостей от другого маршрутизатора, рекламирующего метку для определенного FEC. Каждый маршрутизатор принимает собственное независимое решение о создании метки для каждого FEC. По умолчанию IOS использует независимый элемент управления LSP, в то время как Juniper реализует упорядоченный элемент управления. Оба режима совместимы, разница в том, что упорядоченное управление предотвращает образование черных дыр во время процесса конвергенции LDP за счет замедления самой конвергенции.

Незапрошенный нисходящий поток по сравнению с нисходящим потоком по требованию Нисходящее распределение меток по требованию - это когда LSR должен явно запросить, чтобы метка была отправлена с его нисходящего маршрутизатора для конкретного FEC. Незапрошенная рассылка ярлыков - это когда ярлык отправляется с нижестоящего маршрутизатора без запроса исходного маршрутизатора.

1.8.5.3 **Конфигурация LDP**

mpls ldp

Включить или отключить демон LDP

router-id A.B.C.D

Следующая команда, расположенная в узле маршрутизатора MPLS, настраивает идентификатор маршрутизатора MPLS локального устройства.

ordered-control

Настройте управление распределением этикеток по заказу LDP.

address-family [ipv4 | ipv6]

Настройте LDP для семейства адресов IPv4 или IPv6. Этот подузел, расположенный под узлом маршрута MPLS, позволяет настраивать соседей LDP.

interface IFACE

Расположенная в узле семейства адресов MPLS, используйте эту команду, чтобы включить или отключить обнаружение LDP для каждого интерфейса. IFACE означает имя интерфейса, в

котором включен LDP. По умолчанию он отключен. После выполнения этой команды узел интерфейса семейства адресов настраивается.

discovery transport-address A.B.C.D | A:B::C:D

Расположенная в интерфейсном узле семейства адресов mpls, используйте эту команду, чтобы задать транспортный адрес IPv4 или IPv6, используемый протоколом LDP для взаимодействия с этим интерфейсом.

ttl-security disable

Расположенная под узлом семейства адресов LDP, используйте эту команду, чтобы отключить процедуры GTSM, описанные в RFC 6720 (для семейства адресов IPv4) и RFC 7552 (для семейства адресов IPv6).

Поскольку GTSM является обязательным для LDPv6, единственным результатом отключения GTSM для семейства адресов IPv6 является то, что ldpd не будет отбрасывать пакеты с ограничением перехода ниже 255. Это может быть необходимо для взаимодействия со старыми реализациями. Исходящие пакеты по-прежнему будут отправляться с использованием ограничения перехода 255 для максимальной совместимости.

Если GTSM включен, у соседей с несколькими переходами GTSM должен быть либо отключен индивидуально, либо настроен с соответствующим расстоянием перехода ttl-безопасности.

neighbor A.B.C.D password PASSWORD

Следующая команда, расположенная под узлом маршрутизатора MPLS, настраивает маршрутизатор устройства LDP. Это устройство, если оно найдено, должно соответствовать настроенному паролю. ПАРОЛЬ - это открытый текстовый пароль с его дайджестом, отправляемый по сети.

neighbor A.B.C.D holdtime HOLDTIME

Следующая команда, расположенная под узлом маршрутизатора MPLS, настраивает значение времени ожидания в секундах идентификатора соседа LDP. Его настройка запускает механизм сохранения. Это значение может быть настроено в диапазоне от 15 до 65535 секунд. По истечении этого времени отсутствия ответа сеанс, установленный LDP, будет считаться отключенным. По умолчанию для устройств LDP не настроено время ожидания.

neighbor A.B.C.D ttl-security disable

Расположенная под узлом LDP MPLS, используйте эту команду, чтобы переопределить глобальную конфигурацию и включить / отключить GTSM для указанного соседа.

neighbor A.B.C.D ttl-security hops (1-254)

Расположенная под узлом LDP MPLS, используйте эту команду, чтобы задать максимальное количество переходов, на которое может отсутствовать указанный сосед. Когда GTSM включен для этого соседа, входящие пакеты должны иметь ограничение TTL / hop 256 минус это значение, гарантируя, что они не прошли больше, чем ожидаемое количество переходов. Значение по умолчанию равно 1.

discovery hello holdtime HOLDTIME

discovery hello interval INTERVAL

Значение ИНТЕРВАЛА варьируется от 1 до 65535 секунд. Значение по умолчанию - 5 секунд. Это значение между каждым отправленным сообщением таймера приветствия. Значение времени ожидания варьируется от 1 до 65535 секунд. Значение по умолчанию - 15 секунд. Это значение добавляется как TLV в сообщениях LDP.

dual-stack transport-connectionprefer ipv4

Когда *ldpd* настроен для работы с двумя стеками, предпочтением транспортного соединения по умолчанию является IPv6 (как указано **RFC 7552**). При таких обстоятельствах, *ldpd* откажется устанавливать TCP-соединения через IPv4. Вы можете использовать приведенную выше команду для изменения предпочтения транспортного соединения на IPv4. В этом случае будет возможно распространять сопоставления меток для FEC IPv6 через соединения TCPv4.

1.8.5.4 Показать информацию LDP

Эти команды выводят различные части *ldpd*.

show mpls ldpneighbor[A.B.C.D]

Эта команда сбрасывает различные обнаруженные соседи. Приведенный ниже пример показывает, что у локальной машины есть соседняя операция с идентификатором, установленным в 1.1.1.1.

```
west-vm# show mpls ldp neighbor
AF   ID              State       Remote Address    Uptime
ipv4  1.1.1.1         OPERATIONAL  1.1.1.1          00:01:37
west-vm #
```

show mpls ldp neighbor [A.B.C.D]

show mpls ldp neighbor [A.B.C.D] detail

Приведенные выше команды выводят другую информацию о соседях.

show mpls ldpdiscovery[detail]

show mpls ldp ipv4 discovery[detail]

show mpls ldp ipv6 discovery[detail]

Приведенные выше команды сбрасывают информацию об обнаружении.

show mpls ldp ipv4

show mpls ldp ipv6

Приведенная выше команда сбрасывает интерфейс IPv4 или IPv6 в зависимости от того, где включен LDP. Приведенный ниже вывод иллюстрирует, что сбрасывается для IPv4.

```
west-vm# show mpls ldp ipv4 interface
AF        InterfaceStateUptime   Hello   Timersac
ipv4 eth1       ACTIVE  00:08:35  5/15            0
ipv4 eth3       ACTIVE  00:08:35  5/15            1
```

show mpls ldp ipv4|ipv6

Приведенная выше команда сбрасывает привязку, полученную при обмене MPLS с LDP.

```
west-vm# show mpls ldp ipv4 binding
AF



        1.1.1.1        imp-null    imp-null        no
ipv4  10.115.0.0/24       1.1.1.1        imp-null    17             no
  10.135.0.0/24       1.1.1.1        imp-null    imp-null        no
ipv4  10.200.0.0/24       1.1.1.1        17        imp-null        yes
west-vm#
```

1.8.5.5 Команды отладки LDP

debug mpls ldp KIND

Включить или отключить отладочные сообщения определенного типа. KIND может быть одним из:

discovery
errors
event
labels
messages
zebra

1.8.5.6 **Пример конфигурации**

Приведенная ниже конфигурация дает типичную конфигурацию MPLS устройства, расположенного в магистрали MPLS. LDP включен на двух интерфейсах и попытается подключиться к двум соседям с идентификатором маршрутизатора, установленным на 1.1.1.1 или 3.3.3.3.

```
mpls ldp
 router-id  2.2.2.2
 neighbor  1.1.1.1 password test
 neighbor  3.3.3.3 password test

!
 address-family ipv4
 discovery 2.2.2.2
 !
  interface eth1

!
  interface eth3
!
 exit-address-family
 !
```

Развертывание LDP по магистрали обычно выполняется в топологии конфигурации с полной сеткой . LDP обычно развертывается с помощью IGP, такого как OSPF, который помогает обнаруживать удаленные IP-адреса. Ниже приведен пример извлечения конфигурации OSPF, который поставляется с Конфигурация LDP

```
router ospf
 ospf router-id 2.2.2.2
  network  0.0.0.0/0 area 0
 !
```

Ниже вывод показывает запись маршрута на стороне LER. Запись маршрутизации OSPF (10.200.0.0) связана с записью метки (17) и показывает, что принудительное действие MPLS, направленное на этот адресат, будет применено.

```
north-vm# show ip route
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, E - EIGRP, N - NHRP,
       T - Table, v - VNC, V - VNC-Direct, A - Babel, D - SHARP,
       F - PBR,
       > - selected route, * - FIB route

O>* 1.1.1.1/32 [110/120] via 10.115.0.1, eth2, label 16, 00:00:15
O>* 2.2.2.2/32 [110/20] via 10.115.0.1, eth2, label implicit-null, 00:00:15
O   3.3.3.3/32 [110/10] via 0.0.0.0, loopback1 onlink, 00:01:19
C>* 3.3.3.3/32 is directly connected, loopback1, 00:01:29
O>* 10.0.2.0/24 [110/11] via 10.115.0.1, eth2, label implicit-null, 00:00:15
O   10.100.0.0/24 [110/10] is directly connected, eth1, 00:00:32
C>* 10.100.0.0/24 is directly connected, eth1, 00:00:32
O   10.115.0.0/24 [110/10] is directly connected, eth2, 00:00:25
C>* 10.115.0.0/24 is directly connected, eth2, 00:00:32
O>* 10.135.0.0/24 [110/110] via 10.115.0.1, eth2, label implicit-null, 00:00:15
O>* 10.200.0.0/24 [110/210] via 10.115.0.1, eth2, label 17, 00:00:15
north-vm#
```

Дополнительный пример, демонстрирующий использование некоторых других параметров конфигурации:

```
interface eth0
!
interface eth1
!
interface lo
!
mpls ldp
 dual-stack cisco-interop
 neighbor 10.0.1.5 password opensourcerouting
 neighbor 172.16.0.1 password opensourcerouting
 !
 address-family ipv4
  discovery transport-address 10.0.1.1
  label local advertise explicit-null
  !
  interface eth0
  !
  interface eth1
  !
 !
 address-family ipv6
  discovery transport-address 2001:db8::1
  !
  interface eth1
  !
 !
!
l2vpn ENG type vpls
 bridge br0
 member interface eth2
 !
 member pseudowire mpw0
  neighbor lsr-id 1.1.1.1
  pw-id 100
 !
!
```

1.8.6 **OSPF**

OSPF версии 2 - это протокол маршрутизации, который описан в RFC 2328. OSPF - это IGP. По сравнению с RIP, OSPF может обеспечить масштабируемую сетевую поддержку и более быстрое время конвергенции. OSPF широко используется в крупных сетях, таких как магистральные сети ISP и корпоративные сети.

1.8.6.1 **Основы OSPF**

OSPF - это, в основном, протокол маршрутизации состояния канала. В отличие от протоколов с вектором расстояния, таких как RIP или BGP, где маршрутизаторы описывают доступные пути (т.Е. Маршруты) друг к другу, в протоколах состояния канала маршрутизаторы вместо этого описывают состояние своих соединений с их ближайшими соседними маршрутизаторами.

Каждый маршрутизатор описывает информацию о состоянии своего канала в сообщении, известном как LSA, которое затем передается всем другим маршрутизаторам в домене маршрутизации состояния канала с помощью процесса, называемого потоком. Таким образом, каждый маршрутизатор создает базу данных LSD всех сообщений о состоянии канала. Из этой коллекции LSA в базе данных LSD каждый маршрутизатор может затем вычислить кратчайший путь к любому другому маршрутизатору на основе некоторой общей метрики, используя алгоритм, такой как алгоритм SPF Эдсгера Дейкстры.

Описывая связность сети таким образом, в терминах маршрутизаторов и каналов, а не в терминах путей через сеть, протокол состояния канала может использовать меньшую пропускную способность и сходиться быстрее, чем другие протоколы. Протокол состояния

канала должен распространять только одно сообщение о состоянии канала по всему домену состояния канала, когда канал на любом отдельно взятом маршрутизаторе изменяет состояние, чтобы все маршрутизаторы могли повторно сходиться по лучшим путям через сеть. Напротив, протоколы с вектором расстояния могут требовать последовательности различных сообщений об обновлении пути от ряда разных маршрутизаторов для конвергенции.

Недостатком протокола состояния канала является то, что процесс вычисления наилучших путей может быть относительно интенсивным по сравнению с протоколами с вектором расстояния, в которых почти не требуется никаких вычислений, кроме (потенциально) выбора между несколькими маршрутами. Эти накладные расходы в основном незначительны для современных встроенных процессоров, даже для сетей с тысячами узлов. Основная нагрузка на масштабирование заключается в большей степени в том, чтобы справляться со все большей частотой обновлений LSA по мере увеличения размера области состояния канала, в управлении LSDB и требуемом заполнении.

Цель этого раздела - дать краткое, но точное описание наиболее важных функций OSPF, которые администратору может потребоваться знать, чтобы наилучшим образом настроить OSPF и устранить неполадки.

1.8.6.1.1 Механизмы OSPF

OSPF определяет ряд механизмов, связанных с обнаружением, описанием и распространением состояния по сети. Почти все эти механизмы будут рассмотрены более подробно далее. Они могут быть широко классифицированы как:

1.8.6.1.1.1 OSPF Hello

Протокол OSPF Hello позволяет OSPF быстро обнаруживать изменения в двусторонней доступности между маршрутизаторами в канале. OSPF может дополнительно использовать другие источники информации о доступности, такие как информация о состоянии канала, предоставляемая оборудованием, или через специальные протоколы доступности, такие как BFD.

OSPF также использует протокол Hello для распространения определенного состояния между маршрутизаторами, совместно использующими ссылку, например:

Привет, настроенное состояние протокола, такое как мертвый интервал.

Приоритет маршрутизатора для выбора DR / BDR.

Результаты выборов DR / BDR.

Любые дополнительные возможности, поддерживаемые каждым маршрутизатором.

Протокол Hello сравнительно тривиален и не будет рассмотрен более подробно.

1.8.6.1.1.2 LSA

В основе OSPF лежат сообщения LSA. Несмотря на название, некоторые LSA, строго говоря, не описывают информацию о состоянии канала. Общие LSA описывают такую информацию, как:

Маршрутизаторы, с точки зрения их связей.

Сети, с точки зрения подключенных маршрутизаторов.

Маршруты, внешние по отношению к домену состояния ссылки:

Внешние маршруты

Маршруты, полностью внешние по отношению к OSPF. Маршрутизаторы, инициирующие такие маршруты, известны как маршрутизаторы ASBR.

Сводные маршруты

Маршруты, которые обобщают информацию о маршрутизации, относящуюся к областям OSPF, внешним по отношению к области состояния канала OSPF, созданной маршрутизаторами ABR.

Наводнение LSA

OSPF определяет несколько связанных механизмов, используемых для управления синхронизацией LSDB-ов между соседями, поскольку соседи образуют смежности, а также распространение или затопление новых или обновленных LSA-ов.

1.8.6.1.1.3 AREa

OSPF предусматривает разделение протокола на множество меньших и независимых областей состояния канала. Каждая область должна быть подключена к общей магистральной области с помощью ABR. Эти маршрутизаторы ABR отвечают за обобщение информации о маршрутизации состояния канала в области в сводные LSA, возможно, в сжатой (т. Е. агрегированной) форме, а затем отправляют эти сводки во все другие области, к которым подключен ABR.

Обратите внимание, что между областями передаются только сводки и внешние маршруты. Поскольку они описывают пути, а не какие-либо состояния соединения маршрутизатора, маршрутизация между областями, следовательно, осуществляется по вектору расстояния, а не по состоянию соединения.

1.8.6.1.2 OSPF LSA

Основными объектами в OSPF являются LSA. Все остальное в OSPF вращается вокруг определения того, что описывать в LSA, когда их обновлять, как распространять их по всей сети и как рассчитывать маршруты на их основе.

Существует множество различных LSA-ов для таких целей, как описание фактической информации о состоянии канала, Описание путей (т.Е. Маршрутов), Описание использования полосы пропускания каналов для целей TE и даже произвольных данных посредством непрозрачных LSA-ов.

1.8.6.1.2.1 Заголовок LSA

Все LSA имеют общий заголовок со следующей информацией:

Тип

Разные типы LSA описывают разные вещи в OSPF. Типы включают:

Маршрутизатор LSA

Сетевой LSA

Краткое описание сети LSA

Краткое описание маршрутизатора LSA

AS-Внешний LSA

Ниже рассматриваются особенности различных типов LSA.

Рекламный маршрутизатор

Идентификатор маршрутизатора маршрутизатора, инициирующего LSA.

Идентификатор LSA

Идентификатор LSA, который обычно выводится каким-либо образом из информации, описываемой LSA, например, маршрутизатор LSA использует идентификатор маршрутизатора в качестве идентификатора LSA, сетевой LSA будет иметь IP-адрес DR в качестве идентификатора LSA.

Комбинация типа, идентификатора и идентификатора рекламного маршрутизатора должна однозначно идентифицировать LSA. Однако может существовать несколько

экземпляров LSA с одинаковым типом, идентификатором LSA и идентификатором рекламного маршрутизатора, см. Порядковый номер.

Возраст

Число, позволяющее маршрутизаторам в конечном итоге удалять устаревшие LSA из своих LSDB.

Номинальное значение равно одной из секунд. Возраст 3600, то есть 1 час, называется максимальным. Максимальные LSA игнорируются при вычислениях маршрутизации. LSA должны периодически обновляться их рекламным маршрутизатором до достижения максимального значения, если они должны оставаться действительными.

Маршрутизаторы могут намеренно заполнять LSA с возрастом, искусственно установленным на 3600, чтобы указать, что LSA больше не действителен. Это называется сбросом LSA.

Нет ничего необычного в том, чтобы видеть устаревшие LSA в базе данных LSD, это может произойти, когда маршрутизатор завершает работу без очистки своих LSA (ов), например, когда он отключен от сети. Такие LSA наносят небольшой вред.

Порядковый номер

Число, используемое для отличия новых экземпляров LSA от более старых экземпляров.

1.8.6.1.2.2 LSA

Из всех различных типов LSA-ов только два типа составляют фактическую часть состояния канала OSPF, маршрутизирующие LSA-ы и сетевые LSA-ы. Эти типы LSA являются абсолютно основными для протокола.

Экземпляры этих LSA специфичны для области состояния канала, в которой они созданы. Маршруты, рассчитанные на основе этих двух типов LSA, называются внутрирайонными маршрутами.

Маршрутизатор LSA

Каждый маршрутизатор OSPF должен инициировать LSA маршрутизатора для описания себя. В ней маршрутизатор перечисляет каждый из своих интерфейсов с поддержкой OSPF для данной области состояния канала в терминах:

Стоимость

Стоимость вывода этого интерфейса, обратно пропорциональная некоторому общеизвестному эталонному значению, auto-cost reference-bandwidth (1-4294967).

Тип ссылки

Транзитная сеть

Ссылка на сеть с множественным доступом, в которой маршрутизатор имеет по крайней мере одно полное соединение с другим маршрутизатором.

PtP

Ссылка на один удаленный маршрутизатор с полной смежностью. Для таких ссылок не выбирается DR; для такой ссылки не создается сетевой LSA.

Заглушка

Ссылка без смежных соседей или маршрут хоста.

Идентификатор ссылки и данные

Эти значения зависят от типа ссылки:

Тип ссылки	Идентификатор ссылки	Данные ссылки
Транзит	Ссылка на IP-адрес DR	IP-адрес интерфейса
Двухточечный	Идентификатор удаленного	IP-адрес локального интерфейса

Тип ссылки	Идентификатор ссылки	Данные ссылки
	маршрутизатора	или ifindex для ненумерованных ссылок
Заглушка	IP-адрес	Маска подсети

Ссылки на маршрутизаторе могут быть перечислены несколько раз в LSA маршрутизатора, например, интерфейс PtP, на котором включен OSPF, всегда должен описываться как ссылка-заглушка в LSA маршрутизатора, в дополнение к тому, что он указан как ссылка PtP в LSA маршрутизатора, если смежность с удаленным маршрутизатором полная.

Ссылки-заглушки также могут использоваться как способ описания ссылок, на которых OSPF не используется, известных как пассивные интерфейсы, см. ip ospf passive [A.B.C.D].

Сетевой LSA

В каналах множественного доступа (например, в сетях ethernet'a, некоторых типах ATM и конфигурациях X.25) маршрутизаторы выбирают DR. DR отвечает за создание сетевого LSA, что помогает сократить объем информации, необходимой для описания сетей с множественным доступом с подключением нескольких маршрутизаторов. DR также действует как узел для затопления LSA-ов по этому каналу, тем самым уменьшая накладные расходы на затопление.

Содержимое сетевого LSA описывает:

Маска подсети

Поскольку идентификатор LSA сетевого LSA должен быть IP-адресом DR, маска подсети вместе с идентификатором LSA дает вам сетевой адрес.

Подключенные маршрутизаторы

Каждый маршрутизатор, полностью смежный с DR, указан в LSA по их идентификатору маршрутизатора. Это позволяет легко извлекать соответствующие LSA маршрутизатора из базы данных LSDB.

Краткое описание LSA состояния соединения:

Тип LSA	Идентификатор LSA	Данные LSA описывают
Маршрутизатор LSA	Идентификатор маршрутизатора	OSPF включил ссылки маршрутизатора в пределах определенной области состояния канала.
Сетевой LSA	IP-адрес DR для сети	Маска подсети сети и идентификаторы маршрутизаторов всех маршрутизаторов в сети

С помощью LSDB, состоящей только из этих двух типов LSA, можно построить ориентированный граф связности между всеми маршрутизаторами и сетями в заданной области состояния канала OSPF. Поэтому неудивительно, что, когда маршрутизаторы OSPF создают обновленные таблицы маршрутизации, первый этап вычисления SPF касается только этих двух типов LSA.

1.8.6.1.2.3 Примеры LSA с состоянием ссылки

В приведенном ниже примере показаны два LSA, оба созданные одним и тем же маршрутизатором (идентификатор маршрутизатора 192.168.0.49) и с одинаковым идентификатором LSA (192.168.0.49), но разных типов LSA.

Первый LSA, являющийся LSA маршрутизатора, описывает ссылки 192.168.0.49: 2 ссылки на сети с множественным доступом с полностью смежными соседями (т. Е. Транзитные ссылки) и 1, являющийся заглушкой (без соседних соседей).

Второй LSA является сетевым LSA, для которого 192.168.0.49 является DR, в котором перечислены идентификаторы маршрутизаторов 4 маршрутизаторов в этой сети, которые полностью смежны с 192.168.0.49.

```
# show ip ospf database router 192.168.0.49

       OSPF Router with ID (192.168.0.53)

                Router Link States (Area 0.0.0.0)

  LS age: 38
  Options: 0x2  : *|-|-|-|-|-|E|*
  LS Flags: 0x6
  Flags: 0x2 : ASBR
  LS Type: router-LSA
  Link State ID: 192.168.0.49
  Advertising Router: 192.168.0.49
  LS Seq Number: 80000f90
  Checksum: 0x518b
  Length: 60
   Number of Links: 3

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.1.3
     (Link Data) Router Interface address: 192.168.1.3
      Number of TOS metrics: 0
       TOS 0 Metric: 10

    Link connected to: a Transit Network
     (Link ID) Designated Router address: 192.168.0.49
     (Link Data) Router Interface address: 192.168.0.49
      Number of TOS metrics: 0
       TOS 0 Metric: 10

    Link connected to: Stub Network
     (Link ID) Net: 192.168.3.190
     (Link Data) Network Mask: 255.255.255.255
      Number of TOS metrics: 0
       TOS 0 Metric: 39063
# show ip ospf database network 192.168.0.49

       OSPF Router with ID (192.168.0.53)

                Net Link States (Area 0.0.0.0)

  LS age: 285
  Options: 0x2  : *|-|-|-|-|-|E|*
  LS Flags: 0x6
  LS Type: network-LSA
  Link State ID: 192.168.0.49 (address of Designated Router)
  Advertising Router: 192.168.0.49
  LS Seq Number: 80000074
  Checksum: 0x0103
  Length: 40
  Network Mask: /29
        Attached Router: 192.168.0.49
        Attached Router: 192.168.0.52
        Attached Router: 192.168.0.53
        Attached Router: 192.168.0.54
```

Обратите внимание, что из одного LSA вы можете найти другой. Например, учитывая Network-LSA, у вас есть список идентификаторов маршрутизаторов в этой сети, из которого вы можете затем найти в локальной базе данных LSD соответствующий LSA маршрутизатора. С этого маршрутизатора-LSA вы можете (потенциально) найти ссылки на другие транзитные сети и

идентификаторы маршрутизаторов, которые можно использовать для поиска соответствующего маршрутизатора или сетевого LSA. И таким образом можно найти все маршрутизаторы и сети, доступные из этого начального LSA.

Вместо этого, учитывая LSA маршрутизатора, у вас есть IP-адрес DR любых подключенных транзитных каналов. Сетевые LSA будут иметь этот IP-адрес в качестве идентификатора LSA, поэтому вы можете затем найти этот сетевой LSA и по нему найти все подключенные маршрутизаторы по этому каналу, что потенциально приведет к большему количеству ссылок, сетевых и маршрутизирующих LSA и т. Д. И т. Д.

Только из двух приведенных выше LSA-ов уже можно увидеть следующую частичную топологию:

```
----------------------- Network: ......
            |           Designated Router IP: 192.168.1.3
            |
      IP: 192.168.1.3
       (transit link)
        (cost: 10)
   Router ID: 192.168.0.49(stub)---------- IP: 192.168.3.190/32
        (cost: 10)        (cost: 39063)
       (transit link)
      IP: 192.168.0.49
            |
            |
------------------------------ Network: 192.168.0.48/29
  |         |           |       Designated Router IP: 192.168.0.49
  |         |           |
  |         |     Router ID: 192.168.0.54
  |         |
  |   Router ID: 192.168.0.53
  |
Router ID: 192.168.0.52
```

Обратите внимание, что идентификаторы маршрутизаторов, хотя они выглядят как IP-адреса и часто являются IP-адресами, строго говоря, не являются IP-адресами и не обязательно должны быть доступными адресами (хотя OSPF вычислит маршруты к идентификаторам маршрутизаторов).

1.8.6.1.2.4 Внешние LSA

Внешние, или “Тип 5”, LSA описывают информацию о маршруте, которая полностью является внешней по отношению к OSPF и “вводится” в OSPF. Такая информация о маршрутизации могла поступать из другого протокола маршрутизации, такого как RIP или BGP, они могут представлять статические маршруты или они могут представлять маршрут по умолчанию.

Маршрутизатор OSPF, который генерирует внешние LSA, известен как ASBR. В отличие от LSA-ов с состоянием канала и большинства других LSA-ов, которые затопляются только в пределах области, в которой они происходят, внешние LSA-ы затопляются по всей сети OSPF во все области, способные передавать внешние LSA-ы (области).

Маршруты, внутренние для OSPF (внутри области или между областями), всегда предпочтительнее внешних маршрутов.

Внешний LSA описывает следующее:

Номер IP-сети

Номер IP-сети маршрута описывается полем идентификатора LSA.

Маска IP-сети

Тело внешнего LSA описывает маску IP-сети маршрута. Это, вместе с идентификатором LSA, описывает префикс соответствующего IP-маршрута.

Метрика

Стоимость внешнего маршрута. Эти затраты могут быть затратами OSPF (также известными как показатель “Типа 1”), то есть эквивалентными обычным затратам OSPF, или затратами, полученными извне (показатель “Типа 2”), которые несопоставимы с затратами OSPF и всегда считаются большими, чем любые затраты OSPF. Там, где для маршрута есть как внешние маршруты типа 1, так и 2, тип 1 всегда является предпочтительным.

Адрес пересылки

Адрес маршрутизатора для пересылки пакетов для маршрута. Это может быть и обычно остается равным 0, чтобы указать, что следует использовать ASBR, инициирующий внешний LSA. Для использования адреса пересылки должен существовать внутренний маршрут OSPF к адресу пересылки.

Тег

Произвольный 4-байтовый массив данных, не интерпретируемый OSPF, который может содержать любую информацию о маршруте, которую желают носители OSPF.

1.8.6.1.2.5 В качестве внешнего примера LSA

Для иллюстрации ниже приведен пример внешнего LSA в базе данных LSD маршрутизатора OSPF. Он описывает маршрут к префиксу IP 192.168.165.0 /24, созданный ASBR с идентификатором маршрутизатора 192.168.0.49. Показатель 20 является внешним по отношению к OSPF. Адрес пересылки равен 0, поэтому маршрут должен перенаправляться на исходный ASBR, если он выбран.

```
# show ip ospf database external 192.168.165.0
  LS age:  995
  Options: 0x2  : *|-|-|-|-|-|E |*
  LS : 0x9
  LS : AS-external-LSA
   LinkState ID:  192.168.165.0  (External Network )
  Advertising Router:  192.168.0.49
  LS Seq : 800001d8
  Checksum: 0xea27
  Length:  36
  Network :  / 24
        Metric :  2  (Larger  thanany link  )
        TOS:  0
        Metric:  20
        Forward Address:  0.0.0.0
        External Route :  0
```

Мы можем добавить это к нашей частичной топологии сверху, которая теперь выглядит так:

```
--------------------- Network: ......
         |            Designated Router :  192.168.1.3
         |
   IP: 192.168.1.3      /---- External route:  192.168.165.0/24
     (transit link)     /               Cost:  20  (External metric)
      (cost:  10)       /
Router :  192.168.0.49 (stub)---------- IP: 192.168.3.190/32
      (cost:  10)         (cost:  39063)
     (transit link)
   IP: 192.168.0.49
         |
         |
------------------------------ Network: 192.168.0.48/29
   |        |           |       Designated Router : 192.168.0.49
   |        |           |
   |        |     Router :  192.168.0.54
   |        |
   |   Router :  192.168.0.53
   |
```

```
Router :  192.168.0.52
```

1.8.6.1.2.6 Сводные LSA

Сводные LSA создаются ABR s для обобщения пунктов назначения, доступных в пределах одной области, для других областей. Эти LSA могут описывать IP-сети, потенциально в агрегированной форме, или маршрутизаторы ASBR.

1.8.6.2 Настройка OSPF

Ospfd принимает все распространенные параметры вызова.
-n, --instance

Укажите номер экземпляра для этого вызова *ospfd*.
-a, --apiserver

Включите сервер API OSPF. Это необходимо для использования ospfclient.

ospfd должен получать информацию об интерфейсе от zebra, чтобы функционировать. Поэтому zebra должна быть запущена перед вызовом ospfd. Кроме того, если zebra перезапущен, то ospfd тоже должен быть.

Как и другие демоны, настройка ospfd выполняется в файле конфигурации, специфичном для OSPFospfd.conf, когда встроенная конфигурация не используется.

1.8.6.2.1 Поддержка нескольких экземпляров

OSPF поддерживает несколько экземпляров. Каждый экземпляр идентифицируется положительным ненулевым целым числом, которое необходимо указывать при добавлении элементов конфигурации, специфичных для этого экземпляра. Включение экземпляров выполняется с помощью `/etc/frr/daemons` следующим образом:

```
...
ospfd=yes
ospfd_instances=1,5,6
 ...
```

`ospfd_instances` Переменная определяет, какие экземпляры запускаются и каковы их идентификаторы. В этом примере после запуска FRR вы должны увидеть следующие процессы:

```
# ps -ef | grep "ospfd" frr      11816      1   0  17:30 ?        00:00:00 /usr/lib/frr/ospfd --
daemon -A 127.0.0.1 .0.0.1-n 1

frr      11822      1   0  17:30 ?        00:00:00 /usr/lib/frr/ospfd --daemon -A 127.0.0.1
.0.0.1-n 2
frr      11828      1   0  17:30 ?        00:00:00 /usr/lib/frr/ospfd --daemon -A 127.0.0.1
.0.0.1-n 3
```

Номер экземпляра должен быть указан в конфигурации при обращении к конкретному экземпляру:

```
router ospf 5
   ospf router-id 1.2.3.4
   area  0.0.0.0 authentication message-digest
 ...
```

1.8.6.2.2 Маршрутизаторы

Чтобы запустить процесс OSPF, вы должны указать маршрутизатор OSPF.
router ospf[{(1-65535)|vrf NAME}]

Включить или отключить процесс OSPF.
Несколько экземпляров не поддерживают ИМЯ vrf.
ospf router-idA.B.C.D

Это устанавливает идентификатор маршрутизатора процесса OSPF. Идентификатор маршрутизатора может быть IP-адресом маршрутизатора, но это необязательно - это может быть любое произвольное 32-битное число. Однако он ДОЛЖЕН быть уникальным во всем домене OSPF для динамика OSPF - плохие вещи произойдут, если несколько динамиков OSPF настроены с одним и тем же идентификатором маршрутизатора! Если один не указан, то *ospfd* автоматически получит идентификатор маршрутизатора из *zebra*.

ospf abr-type TYPE

Тип может быть cisco | ibm | shortcut | standard. Типы “Cisco” и “IBM” эквивалентны.

Стандарт OSPF для поведения ABR не позволяет ABR рассматривать маршруты через области, не являющиеся магистральными, когда его соединения с магистралью отключены, даже если в присоединенных областях, не являющихся магистральными, есть другие ABR, которые все еще могут достигать магистрали - это ограничение существует в первую очередь для предотвращения циклов маршрутизации.

С типом ABR “Cisco” или “IBM”, используемым по умолчанию в этой версии FRR, это ограничение снимается, позволяя ABR рассматривать сводки, полученные от других ABR через неосновные области, и, следовательно, маршрутизировать через неосновные области в качестве последнего средства, когда и только когда магистральныйссылки отсутствуют.

Обратите внимание, что области с полностью смежными виртуальными ссылками считаются “транзитными” и всегда могут использоваться для маршрутизации магистрального трафика, и, следовательно, на них не влияет этот параметр (area A.B.C.D virtual-link A.B.C.D).

Более подробную информацию о поведении, управляемом этой командой, можно найти в RFC 3509, и draft-ietf-ospf-shortcut-abr-02.txt .

Цитата: “Хотя определение ABR в спецификации OSPF не требует, чтобы маршрутизатор с несколькими подключенными областями имел магистральное соединение, на самом деле это необходимо для обеспечения успешной маршрутизации между областями и внешними адресатами. Если это требование не выполняется, весь трафик, предназначенный для областей, не подключенных к такому ABR или из домена OSPF, отбрасывается. В этом документе описаны альтернативные варианты поведения ABR, реализованные в маршрутизаторах Cisco и IBM.”

ospf rfc1583compatibility

RFC 2328, преемник RFC 1583, предлагает в соответствии с разделом G.2 (изменения) в разделе 16.4 изменить алгоритм предпочтения пути, который предотвращает возможные циклы маршрутизации, которые были возможны в старой версии OSPFv2. Более конкретно, это требует, чтобы пути между областями и магистральный путь внутри области теперь имели одинаковое предпочтение, но все же оба предпочтительнее внешних путей.

Эта команда не должна быть установлена нормально.

log-adjacency-changes [detail]

Настраивает ospfd для регистрации изменений в смежности. С дополнительным аргументом detail отображаются все изменения в статусе смежности. Без подробностей показаны только изменения в full или регрессии.

passive-interface default

Сделайте все интерфейсы, принадлежащие этому маршрутизатору, пассивными по умолчанию. Описание пассивного интерфейса см. в `ip ospf passive [A.B.C.D]`. Конфигурация для каждого интерфейса имеет приоритет над значением по умолчанию.

timers throttle spf (0-600000) (0-600000) (0-600000)

Эта команда устанавливает начальную задержку, начальное время ожидания и максимальное время ожидания между вычислением SPF и событием, которое

вызвало вычисление. Время указывается в миллисекундах и должно находиться в диапазоне от 0 до 600000 миллисекунд.

Задержка определяет минимальное время задержки вычисления SPF (следовательно, она влияет на то, как долго откладывается вычисление SPF после события, которое происходит за пределами времени ожидания любого предыдущего вычисления SPF, а также служит минимальным временем ожидания).

Последовательные вычисления SPF всегда будут разделены по крайней мере миллисекундами ‘времени ожидания’. Время ожидания является адаптивным и изначально устанавливается на начальное время ожидания, настроенное с помощью приведенной выше команды. События, которые происходят в течение времени ожидания предыдущего вычисления SPF, приведут к увеличению времени ожидания на начальное время ожидания, ограниченное максимальным временем ожидания, настроенным с помощью этой команды. Если адаптивное время ожидания истекает без какого-либо события, запускающего SPF, тогда текущее время ожидания сбрасывается на начальное время ожидания. Текущее время ожидания можно просмотреть спомощью show ip ospf, где оно выражается как множитель начального времени ожидания.

```
router ospf
timers throttle spf 200 400 10000
```

В этом примере задержка установлена на 200 мс, начальное время ожидания установлено на 400 мс, а максимальное время ожидания - на 10 секунд. Следовательно, между событием, требующим вычисления SPF, и фактическим вычислением SPF всегда будет не менее 200 мс. Последующие последовательные вычисления SPF всегда будут разделены промежутком от 400 мс до 10 с, время ожидания увеличивается на 400 мс каждый раз, когда событие, запускающее SPF, происходит в течение времени ожидания предыдущего вычисления SPF.

Эта команда заменяет команду timers spf в предыдущих версиях FRR.

max-metric router-lsa [on-startup (5-86400)|on-shutdown (5-100)]

max-metric router-lsa administrative

Это позволяет поддерживать RFC 3137, где процесс OSPF описывает свои транзитные каналы в своем маршрутизаторе-LSA как имеющие бесконечное расстояние, чтобы другие маршрутизаторы избегали вычисления транзитных путей через маршрутизатор, сохраняя при этом возможность доступа к сетям через маршрутизатор.

Эта поддержка может быть включена административно (и бессрочно) или условно. Условное включение max-metric router-lsa может выполняться в течение нескольких секунд после запуска и / или в течение нескольких секунд до выключения.

Включение этого в течение некоторого времени после запуска позволяет OSPF сначала полностью конвергировать, не затрагивая какие-либо существующие маршруты, используемые другими маршрутизаторами, при этом сохраняя доступность любых подключенных заглушек и / или перераспределенных маршрутов. Включение этого параметра на некоторое время перед завершением работы позволяет маршрутизатору корректно удалиться из домена OSPF.

Административное включение этой функции позволяет осуществлять административное вмешательство по любой причине в течение неопределенного периода времени. Обратите внимание, что если конфигурация записывается в файл, эта административная форма команды stub-router также будет записана в файл. Если ospfd будет перезапущен позже, команда вступит в силу до тех пор, пока не будет изменена вручную.

Настроенное состояние этой функции, а также текущее состояние, например, количество секунд, оставшихся до завершения запуска или выключения, можно просмотреть с show ip ospfпомощью команды.

auto-cost reference-bandwidth (1-4294967)

Это задает базовую полосу пропускания для расчета стоимости, где эта полоса пропускания считается эквивалентной стоимости OSPF, равной 1, указанной в Мбит / с. По умолчанию используется 100 Мбит / с (т.Е. канал с пропускной способностью 100 Мбит / с или выше будет стоить 1. Стоимость каналов с более низкой пропускной способностью будет масштабироваться со ссылкой на эту стоимость).

Этот параметр конфигурации ДОЛЖЕН быть согласован для всех маршрутизаторов в домене OSPF.

network A.B.C.D/M area A.B.C.D

network A.B.C.D/M area (0-4294967295)

Эта команда определяет интерфейсы с поддержкой OSPF. Если интерфейс имеет адрес из диапазона 192.168.1.0 / 24, то приведенная ниже команда включает ospf на этом интерфейсе, чтобы маршрутизатор мог предоставлять сетевую информацию другим маршрутизаторам ospf через этот интерфейс.

```
router ospf
network  192.168.1.0/24 area 0.0.0.0
```

Длина префикса в интерфейсе должна быть равна или больше (т. е. Меньше сети), чем длина префикса в инструкции network. Например, приведенная выше инструкция не включает ospf в интерфейсе с адресом 192.168.1.1 / 23, но это происходит в интерфейсе с адресом 192.168.1.129 / 25.

Обратите внимание, что поведение, когда в интерфейсе определен одноранговый адрес, изменилось после выпуска 0.99.7. В настоящее время, если был настроен одноранговый префикс, мы проверяем, содержит ли префикс в сетевой команде префикс назначения. В противном случае мы проверяем, содержит ли префикс сетевой команды префикс локального адреса интерфейса.

Также можно включить OSPF для каждого интерфейса / подсети с помощью команды интерфейса (ip ospf area AREA [ADDR]). Однако смешивание сетевых команд (network) и интерфейсных команд (ip ospf) на одном маршрутизаторе не поддерживается.

proactive-arp

Эта команда включает или отключает отправку запросов ARP для обновления записей соседней таблицы. Это ускоряет конвергенцию для сетей / 32 по P2P-соединению.

Эта функция включена по умолчанию.

clear ip ospf [(1-65535)] process

Эта команда может использоваться для очистки структур данных процесса ospf. Это также очистит соседство ospf, и оно будет восстановлено. Это также очистит базу данных LSDB. Это будет полезно при изменении идентификатора маршрутизатора, и, если пользователь хочет, чтобы изменение идентификатора маршрутизатора вступило в силу, пользователь может использовать этот cli вместо перезапуска демона ospfd.

clear ip ospf [(1-65535)] neighbor

Эта команда может использоваться для очистки соседних структур данных ospf. Это очистит соседство ospf, и оно будет восстановлено. Эта команда может использоваться, когда соседнее состояние застревает в каком-то состоянии, и это может быть использовано для его восстановления из этого состояния.

maximum-paths (1-64)

Используйте эту команду для управления максимальным количеством путей равной стоимости для достижения определенного назначения. Верхний предел может отличаться, если вы измените значение MULTIPATH_NUM во время компиляции. По умолчанию используется значение MULTIPATH_NUM (64).

write-multiplier (1-100)

Используйте эту команду, чтобы настроить объем работы, выполняемой в потоках чтения и записи пакетов, прежде чем отказаться от управления. Параметр - это количество пакетов, которые необходимо обработать перед возвратом. Значение по умолчанию для этого параметра равно 20.

1.8.6.2.3 Области

area A.B.C.D range A.B.C.D/M [advertise [cost (0-16777215)]]

area (0-4294967295) range A.B.C.D/M [advertise [cost (0-16777215)]]

Сведите пути внутри области из указанной области в одну сводку типа 3-LSA, объявленную для других областей. Эта команда может использоваться только в ABR, и ТОЛЬКО маршрутизаторы-LSA (тип-1) и сетевые LSA (тип-2) (т. Е. LSA с областью видимости) могут быть обобщены. Тип-5 AS-external-LSA не могут быть обобщены - их область действия равна AS . Обобщение типа-7 КАК-external-LSA еще не поддерживается FRR.

```
router ospf
 network  192.168.1.0 /24 area 0.0.0.0
 network  10.0.0.0 / 8 area 0.0.0.10
 area  0.0.0.10 range 10.0.0.0 / 8
```

При конфигурации выше одно краткое описание типа 3-LSA с информацией о маршрутизации 10.0.0.0 / 8 объявляется в магистральную область, если область 0.0.0.10 содержит хотя бы одну внутриобластную сеть (т.Е. Описанную с помощью маршрутизатора или сетевого LSA) из этого диапазона.

area A.B.C.D range A.B.C.D/M not-advertise

area (0-4294967295) range A.B.C.D/M not-advertise

Вместо суммирования путей внутри области фильтруйте их - т. Е. Пути внутри области из этого диапазона не рекламируются в других областях. Эта команда имеет смысл только в ABR.

area A.B.C.D range A.B.C.D/M {substitute A.B.C.D/M|cost (0-16777215)}

area (0-4294967295) range A.B.C.D/M {substitute A.B.C.D/M|cost (0-16777215)}

Замените обобщенный префикс другим префиксом.

```
router ospf
 network 192.168.1.0/24 area 0.0.0.0
 network 10.0.0.0/8 area 0.0.0.10
 area 0.0.0.10 range 10.0.0.0/8 substitute 11.0.0.0/8
```

Одна сводка типа 3-LSA с информацией о маршрутизации 11.0.0.0 / 8 объявляется в магистральную область, если область 0.0.0.10 содержит хотя бы одну внутризоновую сеть (т.Е. Описанную с помощью router-LSA или network-LSA) из диапазона 10.0.0.0 / 8.

По умолчанию показатель суммарного маршрута рассчитывается как наивысший показатель среди суммарных маршрутов. Однако параметр cost можно использовать для задания явной метрики.

Эта команда имеет смысл только в ABR.

area A.B.C.D virtual-link A.B.C.D

area (0-4294967295) virtual-link A.B.C.D

area A.B.C.D shortcut

area (0-4294967295) shortcut

Настройте область в качестве ярлыка. Смотрите **RFC 3509**. Для этого требуется, чтобы для параметра ‘abr-type’ было установлено значение ‘shortcut’.

area A.B.C.D stub

area (0-4294967295) stub

Настройте область как область заглушки. То есть область, в которой ни один маршрутизатор не инициирует маршруты, внешние по отношению к OSPF, и, следовательно, область, в которой все внешние маршруты проходят через ABR (ы). Следовательно, ABR для такой области не нужно передавать в область AS-External LSA (тип-5s) или ASBR-Summary LSA (тип-4). Им нужно только передавать сводные сетевые (типа 3) LSA в такую область вместе со сводкой маршрутов по умолчанию.

area A.B.C.D stub no-summary

area (0-4294967295) stub no-summary

Запрещает *ospfd* ABR вводить сводки между областями в указанную область заглушки.

area A.B.C.D nssa

area (0-4294967295) nssa

Настройте область как NSSA (не очень короткую область). Это область, которая позволяет OSPF импортировать внешние маршруты в область заглушки с помощью нового типа LSA (тип 7). Маршрутизатор границ автономной системы NSSA (ASBR) будет генерировать этот тип LSA. Маршрутизатор границ зоны (ABR) преобразует тип LSA 7 в тип LSA 5, который распространяется в домен OSPF. Области NSSA определены в RFC 3101.

area A.B.C.D nssa suppress-fa

area (0-4294967295) nssa suppress-fa

Настройте маршрутизатор так, чтобы для адреса пересылки было задано значение 0.0.0.0 во всех LSA типа 5, переведенных с LSA типа 7. Маршрутизатор должен быть выбран транслятором области, чтобы эта команда вступила в силу. Эта функция приводит к тому, что маршрутизаторы, настроенные на то, чтобы не объявлять адреса пересылки в магистраль, направляют перенаправленный трафик на транслятор ABR NSSA.

area A.B.C.D default-cost (0-16777215)

Установите стоимость общих LSA по умолчанию, объявленных для коротких областей.

area A.B.C.D export-list NAME

area (0-4294967295) export-list NAME

Сводка типа фильтра-3-LSA, объявленные для других областей, получены из внутризоновых путей из указанной области.

```
router ospf
 network 192.168.1.0/24 area 0.0.0.0
 network 10.0.0.0/8 area 0.0.0.10
 area 0.0.0.10 export-list foo
!
access-list foo permit 10.10.0.0/16
access-list foo deny any
```

С приведенным выше примером любые пути внутри области из области 0.0.0.10 и из диапазона 10.10.0.0 / 16 (например, 10.10.1.0 / 24 и 10.10.2.128 / 30) объявляются в других областях как сводные LSA типа 3, но любые другие (например, 10.11.0.0 / 16 или 10.128.30.16 / 30) не являются.

Эта команда актуальна только в том случае, если маршрутизатор является ABR для указанной области.

area A.B.C.D import-list NAME

area (0-4294967295) import-list NAME

То же, что и export-list, но применяется к путям, объявленным в указанной области как сводные LSA типа 3.

area A.B.C.D filter-list prefix NAME in

area A.B.C.D filter-list prefix NAME out

area (0-4294967295) filter-list prefix NAME in

area (0-4294967295) filter-list prefix NAME out

Фильтрация сводки типа 3-LSA в / из области с использованием списков префиксов. Эта команда имеет смысл только в ABR.

area A.B.C.D authentication

area (0-4294967295) authentication

Укажите, что для данной области должна использоваться простая аутентификация по паролю.

area A.B.C.D authentication message-digest

area (0-4294967295) authentication message-digest

Укажите, что пакеты OSPF должны быть аутентифицированы с помощью MD5 HMAC в пределах заданной области. Вводный материал также должен быть настроен для каждого интерфейса (ip ospf message-digest-key).

Аутентификация MD5 также может быть настроена для каждого интерфейса (ip ospf authentication message-digest). Такие настройки для каждого интерфейса переопределяют любые настройки аутентификации для каждой области.

1.8.6.2.4 Интерфейсы

ip ospf areaAREA[ADDR]

Включите OSPF в интерфейсе, необязательно ограниченном только IP-адресом, указанным ADDR, поместив ее в ОБЛАСТЬ область. Если у вас много интерфейсов и / или много подсетей, включите OSPF с помощью этой команды вместо (network A.B.C.D/M area A.B.C.D) может привести к небольшому улучшению производительности.

Обратите внимание, что при смешивании обеих сетевых команд (network) и команды интерфейса (ip ospf) на том же маршрутизаторе не поддерживается. Если (ip ospf) присутствует, (network) команды завершатся ошибкой.

ip ospf authentication-key AUTH_KEY

Установите ключ аутентификации OSPF на простой пароль. После настройки AUTH_KEYвсе пакеты OSPF аутентифицируются. AUTH_KEY имеет длину до 8 символов.

Простая текстовая аутентификация по паролю небезопасна и устарела в пользу аутентификации MD5 HMAC.

ip ospf authenticationmessage-digest

Укажите, что в этом интерфейсе должна использоваться аутентификация MD5 HMAC. Материал для ввода ключей MD5 также должен быть настроен. Переопределяет любую аутентификацию, включенную для каждой области (area A.B.C.D authentication message-digest)

Обратите внимание, что проверка подлинности OSPF MD5 требует, чтобы время никогда не возвращалось назад (правильное время НЕ важно, важно только, чтобы оно никогда не возвращалось назад), даже при сбросе, если ospfd должен иметь возможность быстро восстанавливать смежность со своими соседями после перезапуска / перезагрузки. При загрузке с внешнего или энергонезависимого источника (например, часы с батарейным питанием, NTP и т. Д.) На хосте должно быть установлено системное время, или же системные часы должны периодически сохраняться в энергонезависимом хранилище и восстанавливаться при загрузке, если ожидается, что аутентификация MD5 будет работать надежно.

ip ospf message-digest-key KEYID md5 KEY

Установите ключ аутентификации OSPF на криптографический пароль. Криптографический алгоритм - MD5.

KEYID определяет секретный ключ, используемый для создания дайджеста сообщения. Этот идентификатор является частью протокола и должен быть согласован между маршрутизаторами в канале.

КЛЮЧ - это фактический ключ дайджеста сообщения, содержащий до 16 символов (строки большего размера будут усечены), и он связан с заданным идентификатором ключа.

ip ospf cost (1-65535)

Установите стоимость ссылки для указанного интерфейса. Значение стоимости устанавливается в поле показателя маршрутизатора-LSA и используется для расчета SPF.

ip ospf dead-interval (1-65535)

ip ospf dead-interval minimal hello-multiplier (2-20)

Установите количество секунд для значения таймера RouterDeadInterval, используемого для таймера ожидания и таймера бездействия. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию равно 40 секундам.

Если вместо этого указано "минимальное", то интервал ожидания устанавливается равным 1 секунде, и необходимо указать множитель приветствия. Множитель приветственных сообщений определяет, сколько приветственных сообщений отправлять в секунду, от 2 (каждые 500 мс) до 20 (каждые 50 мс). Таким образом, для OSPF может быть время сходимости 1 сек. Если указана эта форма, то интервал приветствия, объявленный в пакетах приветствия, устанавливается равным 0, а интервал приветствия для принятых пакетов приветствия не проверяется, таким образом, множитель приветствия НЕ обязательно должен быть одинаковым для нескольких маршрутизаторов в общем канале.

ip ospf hello-interval (1-65535)

Установите количество секунд для значения таймера HelloInterval. При установке этого значения приветственный пакет будет отправляться каждые секунды значения таймера на указанном интерфейсе. Это значение должно быть одинаковым для всех маршрутизаторов, подключенных к общей сети. Значение по умолчанию равно 10 секундам.

Эта команда не имеет никакого эффекта, если ip ospf dead-interval minimal hello-multiplier (2-20) также указана для интерфейса.

ip ospf network (broadcast|non-broadcast|point-to-multipoint|point-to-point [dmvpn])

При настройке двухточечной сети на интерфейсе, а интерфейс имеет адрес /32, связанный с then, OSPF будет рассматривать интерфейс как ненумерованный. Если вы делаете это, вы должны установить net.ipv4.conf.<имя интерфейса>.rp_filter значение равно 0. Для того, чтобы многоадресные пакеты ospf доставлялись ядром.

При использовании в сети DMVPN на спице этот OSPF будет настроен в режиме "точка-точка", но КОНЦЕНТРАТОР будет многоточечным. Чтобы заставить эту топологию работать, укажите необязательный параметр ‘dmvpn’ в спице.

Явно задайте тип сети для указанного интерфейса.

ip ospf priority (0-255)

Установите целочисленное значение RouterPriority. Маршрутизатор с наивысшим приоритетом будет иметь больше прав на назначение назначенным маршрутизатором. Установка значения в 0 делает маршрутизатор не имеющим права становиться назначенным маршрутизатором. Значение по умолчанию равно 1.

ip ospf retransmit-interval (1-65535)

Установите количество секунд для значения таймера RxmtInterval. Это значение используется при повторной передаче пакетов запроса описания базы данных и состояния ссылки. Значение по умолчанию равно 5 секундам.

ip ospf transmit-delay (1-65535) [A.B.C.D]

Установите количество секунд для значения InfTransDelay. Возраст LSA должен быть увеличен на это значение при передаче. Значение по умолчанию равно 1 секунде.

ip ospf passive [A.B.C.D]

Не используйте OSPF в интерфейсе, но рекламируйте интерфейс как заглушку в маршрутизаторе-LSA для этого маршрутизатора. Это позволяет рекламировать адреса на таких подключенных интерфейсах без необходимости создавать LSA AS-External / Type-5 (которые имеют глобальную область заливки), как это произошло бы, если бы подключенные адреса были перераспределены в OSPF (перераспределение). Это единственный способ рекламировать ссылки, не относящиеся к OSPF, в областях-заглушках.

ip ospf area (A.B.C.D|(0-4294967295))

Включите ospf в интерфейсе и задайте соответствующую область.

1.8.6.3 **Карта маршрута OSPF**

Использование *поддержки* карты маршрутов ospfd.

set metric [+|-](0-4294967295)

Установите метрику для согласованного маршрута при отправке объявления. Используйте знак плюс (+), чтобы добавить значение показателя к существующей метрике. Используйте знак минус (-) для вычитания значения метрики из существующей метрики.

Перераспределение

redistribute <babel|bgp|connected|eigrp|isis|kernel|openfabric|ospf|rip|sharp|static|table> [metric -type (1-2)] [metric (0-16777214)] [route-map]

Перераспределите маршруты указанного протокола или типа в OSPF с типом метрики и набором метрик, если указано, фильтруя маршруты с использованием заданной карты маршрутов, если указано. Распространяемые маршруты также могут быть отфильтрованы с помощью списков рассылки, см. Раздел Конфигурация списка рассылки ospf.

Перераспределенные маршруты распределяются в OSPF как внешние LSA типа 5 в ссылки на области, которые принимают внешние маршруты, внешние LSA типа 7 для областей NSSA и вообще не перераспределяются в области-заглушки, где внешние маршруты не разрешены.

Обратите внимание, что для подключенных маршрутов вместо этого можно использовать ip ospf passive [A.B.C.D] конфигурация.

default-information

default-information (0-16777214)

default-information originate metric (0-16777214) metric-type (1/2)

default-information originatemetric (0-16777214) metric-type (1/2) route-map WORD

default-information originate

default-information originate alwaysmetric (0-16777214)

default-information originate alwaysmetric (0-16777214) metric-type (1/2)

default-information originatealwaysmetric (0-16777214) metric-type (1/2) route-map

Создайте КАК внешний (тип 5) LSA, описывающий маршрут по умолчанию во все области, поддерживающие внешнюю маршрутизацию, с указанной метрикой и типом метрики. Если задано ключевое слово ‘always’, то значение по умолчанию объявляется всегда, даже если в таблице маршрутизации нет значения по умолчанию.

distribute-

list NAME out <kernel|connected|static|rip|isis|bgp|eigrp|nhrp|table|vnc|babel|openfabric>

Примените фильтр списка доступа NAME к перераспределяемым маршрутам данного типа, прежде чем разрешить перераспределение маршрутов в OSPF (перераспределение ospf).

default-metric (0-16777214)

distance (1-255)

distance ospf (intra-area|inter-area|external) (1-255)

1.8.6.4 Плавный перезапуск

graceful-restart [grace-period (1-1800)]

Настройте поддержку перезапуска Graceful Restart (RFC 3623). При включении льготный период по умолчанию составляет 120 секунд.

Чтобы выполнить корректное завершение работы, перед перезапуском демона ospfd необходимо выполнить команду уровня EXEC “graceful-restart prepare ip ospf”.

graceful-restart helperenable [A.B.C.D]

Настройте вспомогательную поддержку Graceful Restart (RFC 3623). По умолчанию вспомогательная поддержка отключена для всех соседей. Эта конфигурация включает / отключает вспомогательную поддержку на этом маршрутизаторе для всех соседей. Чтобы включить / отключить вспомогательную поддержку для определенного соседа, должен быть указан идентификатор маршрутизатора (A.B.C.D).

graceful-restart helper strict-lsa-checking

Если настроена ‘strict-lsa-checking’, то помощник прервет корректный перезапуск при изменении LSA, что повлияет на перезапуск маршрутизатора. По умолчанию включена ”строгая проверка lsa""

graceful-restart helpersupported-grace-time

Поддерживается в качестве ВСПОМОГАТЕЛЬНОГО средства для настроенного льготного периода.

graceful-restart helper planned-only

Это помогает поддерживать в качестве ПОМОЩНИКА только для запланированных перезапусков. По умолчанию он поддерживает как запланированные, так и незапланированные отключения.

graceful-restart prepare ip ospf

Инициируйте плавный перезапуск для всех экземпляров OSPF, настроенных с помощью команды “graceful-restart”. Демон ospfd должен быть перезапущен в течение льготного периода для конкретного экземпляра, в противном случае корректный перезапуск завершится неудачей.

Это команда уровня EXEC.

1.8.6.5 Отображение информации

show ip ospf [vrf <NAME|all>] [json]

Отображение информации о различных общих состояниях OSPF и области и информации о конфигурации.

show ip ospf interface[INTERFACE] [json]

Отображение состояния и конфигурации OSPF указанного интерфейса или всех интерфейсов, если интерфейс не указан.

show ip ospf neighbor[json]

show ip ospf neighborINTERFACE[json]

show ip ospf neighbordetail[json]

show ip ospf neighbor A.B.C.D [detail] [json]

show ip ospf neighborINTERFACEdetail[json]

Отображение информации lsa из базы данных LSDB. Json o / p этой команды охватывает базовую информацию о маршруте, то есть все LSA, кроме непрозрачной информации lsa.

show ip ospf [vrf <NAME|all>] database [json]

show ip ospf [vrf <NAME|all>] database(asbr-summary|external|network|router|summary) [json]

show ip ospf [vrf <NAME|all>] database(asbr-summary|external|network|router|summary) LINK-STATE-ID [json]

show ip ospf [vrf <NAME|all>] database(asbr-summary|external|network|router|summary) LINK-STATE-ID adv-router ADV-ROUTER [json]

show ip ospf [vrf <NAME|all>] database(asbr-summary|external|network|router|summary) adv-router ADV-ROUTER [json]

show ip ospf [vrf <NAME|all>] database(asbr-summary|external|network|router|summary) LINK-STATE-ID self-originate [json]

show ip ospf [vrf <NAME|all>] database (asbr-summary|external|network|router|summary) self-originate [json]

show ip ospf [vrf <NAME|all>] database max-age [json]

show ip ospf [vrf <NAME|all>] database self-originate [json]

Показать сводку базы данных OSPF.

show ip ospf route [json]

Показать таблицу маршрутизации OSPF, как определено самым последним вычислением SPF.

show ip ospf [vrf <NAME|all>] border-routers [json]

Показать список сводки пограничных маршрутизаторов ABR и ASBR, полученных с помощью OSPFv2 Type-3 (сводная LSA) и Type-4 (сводная ASBR LSA). Пользователь может получить эту информацию в формате JSON, когда `json` представлено ключевое слово в конце cli.

show ip ospf graceful-restart helper [detail] [json]

Отображает подробные сведения о помощнике перезапуска Grcaeful, включая изменения конфигурации помощника.

1.8.6.6 Непрозрачный LSA

ospf opaque-lsa

capability

ospfd поддерживает непрозрачный LSA (**RFC 2370**) в качестве частичной поддержки MPLS Traffic Engineering LSA. В конфигурации должна быть включена функция opaque-lsa. Альтернативной командой может быть “mpls-te on” (Проектирование дорожного движения). Обратите внимание, что FRR предлагает только частичную поддержку некоторых

расширений протокола маршрутизации, которые используются с MPLS-TE; он не поддерживает полное решение RSVP-TE.

show ip ospf [vrf <NAME|all>] database(opaque-link|opaque-area|opaque-external)

show ip ospf [vrf <NAME|all>] database (opaque-link|opaque-area|opaque-external) LINK-STATE-ID

show ip ospf [vrf <NAME|all>] database(opaque-link|opaque-area|opaque-external) LINK-STATE-ID adv-router ADV-ROUTER

show ip ospf [vrf <NAME|all>] database(opaque-link|opaque-area|opaque-external) adv-router ADV-ROUTER

show ip ospf [vrf <NAME|all>] database(opaque-link|opaque-area|opaque-external) LINK-STATE-ID self-originate

show ip ospf [vrf <NAME|all>] database(opaque-link|opaque-area|opaque-external) self-originate

Показать непрозрачный LSA из базы данных.

show ip ospf (1-65535) reachable-routers

show ip ospf [vrf <NAME|all>] reachable-routers

Показать таблицу маршрутизации доступных маршрутизаторов.

1.8.6.7 Traffic Engineering

В настоящее время FRR предлагает частичную поддержку некоторых расширений протокола маршрутизации, которые могут использоваться с MPLS-TE. В настоящее время FRR не поддерживает полное решение RSVP-TE.

mpls-te on

Включить затопление LSA для управления трафиком.

mpls-te router-address <A.B.C.D>

Настройте стабильный IP-адрес для MPLS-TE. Затем этот IP-адрес объявляется в непрозрачном типе LSA-10 TLV = 1 (TE) вариант 1 (адрес маршрутизатора).

mpls-te inter-as area <area-id>|as

Включить RFC 5392 поддержка - Inter-AS TE v2 - для заполнения инженерных параметров трафика канала Inter-AS. поддерживаются 2 режима: AREA и AS; LSA заливаются в AREA <area-id> с непрозрачным типом-10, соответственно в AS с непрозрачным типом-11. Во всех случаях непрозрачный-LSA TLV = 6.

mpls-te

Экспортируйте базу данных управления трафиком другим демонам с помощью непрозрачных сообщений о состоянии ссылки ZAPI.

show ip ospf mpls-te

show ip ospf mpls-te interface INTERFACE

Отображение параметров управления трафиком MPLS для всего или указанного интерфейса.

show ip ospf mpls-te router

Показать параметры маршрутизатора для управления трафиком.

show ip ospf mpls-te database [verbose|json]

show ip ospf mpls-te database vertex [self-originate|adv-router ADV-ROUTER] [verbose|json]

show ip ospf mpls-te database edge [A.B.C.D] [verbose|json]

show ip ospf mpls-te database subnet [A.B.C.D/M] [verbose|json]

Показать базу данных Traffic Engineering

1.8.6.8 Информация о маршрутизаторе

router-info [as | area]

Включить информацию о маршрутизаторе (**RFC 4970**) Объявление LSA с заполнением области AS (по умолчанию) или области области, когда указана область. Старый синтаксис *область информации о маршрутизаторе <A.B.C.D>* всегда поддерживается, но помечается как устаревший, поскольку идентификатор области больше не нужен. Действительно, информация о маршрутизаторе поддерживает многозональность и автоматически определяет области.

pce address<A.B.C.D>

pce domainas (0-65535)

pce neighboras (0-65535)

pce

pce

Команды соответствуют **RFC 5088** и разрешить маршрутизатору OSPF объявлять возможности элемента вычисления пути (PCE) через LSA информации о маршрутизаторе (RI). Информация о маршрутизаторе должна быть включена до этого. Команда устанавливает / отменяет соответственно IP-адрес PCE, номера автономной системы (AS) контролируемых доменов, соседний ASs, флаг и область действия. Для определения флага и области применения, пожалуйста, обратитесь к: rfc `5088` для распознавания битовых шаблонов. Для указания всех соседей PCE можно указать несколько команд ‘pce neighbor’.

show ip ospf router-info

Показывать флаг возможностей маршрутизатора.

show ip ospf router-info pce

Показать параметры PCE возможностей маршрутизатора.

1.8.6.9 Маршрутизация сегментов

Это ЭКСПЕРИМЕНТАЛЬНАЯ поддержка маршрутизации сегментов в соответствии с RFC 8665 для MPLS dataplane.

segment-routing on

Включить маршрутизацию сегментов. Даже если это также активирует поддержку информации о маршрутизации, предпочтительно также активировать информацию о маршрутизации и соответствующим образом установить область или КАК затопление.

segment-routing global-block (16-1048575) (16-1048575) [local-block (16-1048575) (16-1048575)]

Установите глобальный блок маршрутизации сегмента, т.Е. Диапазон меток, используемый MPLS для хранения метки в MPLS FIB для префикса SID. При необходимости также задайте локальный блок, то есть диапазон меток, используемый для SID смежности. Отрицательная версия команды всегда отключает оба диапазона.

segment-routing node-msd (1-16)

Исправьте максимальную глубину стека, поддерживаемую маршрутизатором. Значение зависит от плана данных MPLS. Например, для ядра Linux, начиная с версии 4.13, оно равно 32.

segment-routing prefixA.B.C.D/M [index (0-65535)|no-php-flag|explicit-null]

В настоящее время поддерживается префикс с /32, соответствующий интерфейсу обратной связи. ‘No-php-flag’ означает ОТСУТСТВИЕ появления предпоследнего перехода, что позволяет узлу SR запрашивать у своего соседа, чтобы он не вставлял метку. ‘explicit-null’ означает, что соседние узлы должны поменять входящую метку на метку MPLS Explicit Null перед доставкой пакета.

show ip ospf databasesegment-routing <adv-router ADVROUTER|self-originate> [json]

Показать базу данных маршрутизации сегмента, все узлы SR, конкретный рекламируемый маршрутизатор или собственный маршрутизатор. Необязательный вывод JSON можно получить, добавив ‘json’ в конец команды.

1.8.6.10 **СумМаризация внешнего маршрута**

Эта функция суммирует исходные внешние LSA (тип-5 и Тип-7). Сводный маршрут будет создан от имени всех сопоставленных внешних LSA.

summary-address A.B.C.D/M [tag (1-4294967295)]

Эта команда включает / отключает суммирование для настроенного диапазона адресов. Тег является необязательным параметром. Если тег настроен, итоговый маршрут будет создан с помощью настроенного тега.

summary-address A.B.C.D/M no-advertise

Эта команда гарантирует, что итоговый lsa не будет отображаться для сопоставленных внешних LSA.

aggregation (5-1800)

Настройте интервал таймера задержки агрегации. Подведение итогов начинается только после истечения этого таймера задержки. По умолчанию интервал задержки составляет 5 секунд.

Форма no команды сбрасывает интервал задержки агрегации до значения по умолчанию.

show ip ospf [vrf <NAME|all>] summary-address [detail] [json]

Показать конфигурацию для отображения всех настроенных сводных маршрутов с соответствующей внешней информацией LSA.

1.8.6.11 **TI-LFA**

Экспериментальная поддержка независимого от топологии LFA (альтернатива без циклов), см., например, ‘проект-bashandy-rtgwg-сегмент-маршрутизация-ti-lfa-05’. Обратите внимание, что TI-LFA требует правильной конфигурации маршрутизации сегментов.

fast-reroute ti-lfa [node-protection]

Настроен на уровне маршрутизатора. Активирует TI-LFA для всех интерфейсов.
Обратите внимание, что пока поддерживаются только интерфейсы P2P.

1.8.6.12 **Отладка OSPF**

debug ospf [(1-65535)] bfd

Включить или отключить отладку для событий BFD. При этом будут показаны сообщения библиотеки интеграции BFD и сообщения интеграции OSPF с BFD, которые в основном связаны с переходами состояний и проблемами проверки.

debug ospf [(1-65535)] client-api

Отображение отладочной информации для OSPF opaque data client API.

debug ospf [(1-65535)] default-information

Показать информацию об отладке информации по умолчанию

debug ospf [(1-65535)] packet(hello|dd|ls-request|ls-update|ls-ack|all) (send|recv) [detail]

Дамп-пакет для отладки
debug ospf [(1-65535)] ism [status|events|timers]

Показать отладочную информацию конечного автомата интерфейса
debug ospf [(1-65535)] nsm [status|events|timers]

Отображение отладочной информации конечного сетевого автомата
debug ospf [(1-65535)]

Отображение отладочной информации о событии OSPF
debug ospf [(1-65535)] nssa

Показать отладочную информацию о не очень заглушенной области
debug ospf [(1-65535)] ldp-sync

Отображение отладочной информации о LDP-синхронизации
debug ospf [(1-65535)] lsa [aggregate|flooding|generate|install|refresh]

Показать сведения об отладке сообщений о состоянии канала
debug ospf [(1-65535)] sr

Отображение отладочной информации о маршрутизации сегментов
debug ospf [(1-65535)] te

Показать отладочную информацию о Traffic Engineering LSA
debug ospf [(1-65535)] ti-lfa

Показать отладочную информацию о SR TI-LFA
debug ospf [(1-65535)] zebra [interface|redistribute]

Отображение отладочной информации API ZEBRA
debug ospf [(1-65535)] graceful-restart

Включение / отключение информации об отладке для OSPF Graceful Restart Helper
show debugging ospf

1.8.6.13 **Пример конфигурации**

Простой пример с включенной аутентификацией MD5:

```
!
! bge0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 ABCDEFGHIJK
!
router ospf
 network  192.168.0.0/16 area 0.0.0.1
 area 0.0.0.1 authentication message-digest
```

An ABR маршрутизатор с аутентификацией MD5 и выполнением обобщения сетей между областями:

```
!
! ABCDEF
log file /var/log/frr/ospfd.log
service advanced-vty

!
interface eth0
 ip ospf authentication message-digest
 ip ospf message-digest-key 1 md5 ABCDEFGHIJK

!
interface ppp0
 ip ospf passive

!
```

```
interface br0
 ip ospf authentication message-digest
 ip ospf message-digest-key 2 md5 XYZ12345

!
router ospf
 ospf router-id 192.168.0.1
 redistribute connected
 network  192.168.0.0 / 24 area 0.0.0.0
 network  10.0.0.0/ 16 area 0.0.0.0
 network  192.168.1.0 / 24 area 0.0.0.1
 area  0.0.0.0 authentication message-digest
 area 0.0.0.0 range 10.0.0.0 / 16
 area  0.0.0.0 range 192.168.0.0 / 24
 area  0.0.0.1 authentication message-digest
 area 0.0.0.1 range 10.2.0.0/ 16
!
```

Конфигурация управления трафиком с поддержкой Inter-ASv2.

Во-первых, zebra.conf часть:

```
interface eth0
 ip address 198.168.1.1/24
 link-params
  enable
  admin-grp 0xa1
  metric 100
  max-bw 1.25e+07
  max-rsv-bw 1.25e+06
  unrsv-bw 0 1.25e+06
  unrsv-bw 1 1.25e+06
  unrsv-bw 2 1.25e+06
  unrsv-bw 3 1.25e+06
  unrsv-bw 4 1.25e+06
  unrsv-bw 5 1.25e+06
  unrsv-bw 6 1.25e+06
  unrsv-bw 7 1.25e+06
!
interface eth1
 ip address 192.168.2.1/24
 link-params
  enable
  metric 10
  max-bw 1.25e+07
  max-rsv-bw 1.25e+06
  unrsv-bw 0 1.25e+06
  unrsv-bw 1 1.25e+06
  unrsv-bw 2 1.25e+06
  unrsv-bw 3 1.25e+06
  unrsv-bw 4 1.25e+06
  unrsv-bw 5 1.25e+06
  unrsv-bw 6 1.25e+06
  unrsv-bw 7 1.25e+06
  neighbor 192.168.2.2 as 65000
   hostname HOSTNAME
   password PASSWORD
   log file /var/log/zebra.log
   !
   interface eth0
    ip address 198.168.1.1/24
    link-params
     enable
     admin-grp 0xa1
     metric 100
     max-bw 1.25e+07
     max-rsv-bw 1.25e+06
     unrsv-bw 0 1.25e+06
     unrsv-bw 1 1.25e+06
     unrsv-bw 2 1.25e+06
     unrsv-bw 3 1.25e+06
```

```
     unrsv-bw 4 1.25e+06
     unrsv-bw 5 1.25e+06
     unrsv-bw 6 1.25e+06
     unrsv-bw 7 1.25e+06
   !
   interface eth1
    ip address 192.168.2.1/24
    link-params
     enable
     metric 10
     max-bw 1.25e+07
     max-rsv-bw 1.25e+06
     unrsv-bw 0 1.25e+06
     unrsv-bw 1 1.25e+06
     unrsv-bw 2 1.25e+06
     unrsv-bw 3 1.25e+06
     unrsv-bw 4 1.25e+06
     unrsv-bw 5 1.25e+06
     unrsv-bw 6 1.25e+06
     unrsv-bw 7 1.25e+06
     neighbor 192.168.2.2 as 65000
```

Затем `ospfd.conf` сам:

```
hostname HOSTNAME
password PASSWORD
log file /var/log/ospfd.log
!
!
interface eth0
 ip ospf hello-interval 60
 ip ospf dead-interval 240
!
interface eth1
 ip ospf hello-interval 60
 ip ospf dead-interval 240
!
!
router ospf
 ospf router-id 192.168.1.1
 network 192.168.0.0/16 area 1
 ospf opaque-lsa
 mpls-te
 mpls-te router-address 192.168.1.1
 mpls-te inter-as area 1
!
```

Пример информации о маршрутизаторе с рекламой PCE:

```
!
router ospf
 ospf router-id 192.168.1.1
 network 192.168.0.0/16 area 1
 capability opaque
 mpls-te
 mpls-te router-address 192.168.1.1
 router-info area 0.0.0.1
 pce address 192.168.1.1
 pce flag 0x80
 pce domain as 65400
 pce neighbor as 65500
 pce neighbor as 65200
 pce scope 0x80
!
```

1.8.7 **VRRP**

VRRP определяет протокол выбора, который динамически назначает ответственность за виртуальный маршрутизатор одному из VRRP-маршрутизаторов в локальной сети. Маршрутизатор VRRP, управляющий адресами IPv4 или IPv6, связанными с виртуальным

маршрутизатором, называется ведущим, и он пересылает пакеты, отправленные на эти адреса IPv4 или IPv6. Основные маршрутизаторы VRRP настроены с виртуальными адресами IPv4 или IPv6, а резервные маршрутизаторы VRRP определяют семейство передаваемых виртуальных адресов на основе транспортного протокола. В маршрутизаторе VRRP виртуальные маршрутизаторы в каждом из семейств адресов IPv4 и IPv6 являются отдельным доменом и не перекрываются. Процесс выбора обеспечивает динамический переход на другой ресурс в ответственности за пересылку, если главный сервер становится недоступен. Для IPv4 преимущество использования VRRP заключается в более высоком уровне доступности пути по умолчанию, не требующем настройки динамической маршрутизации или протоколов обнаружения маршрутизатора на каждом конечном узле. Для IPv6 преимущество использования VRRP для IPv6 заключается в более быстром переключении на резервные маршрутизаторы, чем это может быть достигнуто с помощью стандартных механизмов обнаружения соседей IPv6.

1.8.7.1 Настройка VRRP

vrrp (1-255) [version (2-3)]

Создайте маршрутизатор VRRP с указанным VRID в интерфейсе. Необязательно укажите версию протокола. Если версия протокола не указана, по умолчанию используется VRRPv3

vrrp (1-255) advertisement-interval (10-40950)

Установите интервал рекламы. Это интервал, с которым будут отправляться рекламные объявления VRRP. Значения указаны в миллисекундах, но должны быть кратны 10, поскольку сам VRRP использует сантисекунды.

vrrp (1-255) ip A.B.C.D

Добавьте IPv4-адрес к маршрутизатору. Этот адрес уже должен быть настроен на соответствующем устройстве macvlan. Добавление IP-адреса к маршрутизатору неявно активирует маршрутизатор; смотрите `[no] vrrp (1-255) shutdown` как переопределить это поведение.

vrrp (1-255) ipv6 X:X::X:X

Добавьте IPv6-адрес к маршрутизатору. Этот адрес уже должен быть настроен на соответствующем устройстве macvlan. Добавление IP-адреса к маршрутизатору неявно активирует маршрутизатор; смотрите [no] vrrp (1-255) shutdown как переопределить это поведение.

Эта команда завершится ошибкой, если для версии протокола установлено значение VRRPv2, поскольку VRRPv2 не поддерживает IPv6.

vrrp (1-255) preempt

Переключите режим вытеснения. При включении вытеснение позволяет резервным маршрутизаторам с более высоким приоритетом перенять статус Master у существующего Master. Включено по умолчанию.

vrrp (1-255) checksum-with-ipv4-pseudoheader

кажите, должна ли контрольная сумма VRRPv3 включать псевдоним IPv4. Эта команда не должна влиять на VRRPv2 и IPv6. Включена по умолчанию

vrrp (1-255) priority (1-254)

Установите приоритет маршрутизатора. Маршрутизатор с наивысшим приоритетом выбирается в качестве ведущего. Если все маршрутизаторы в виртуальном маршрутизаторе VRRP настроены с одинаковым приоритетом, маршрутизатор с наивысшим основным IP-адресом выбирается в качестве ведущего. Значение приоритета 255 зарезервировано для действующего главного маршрутизатора.

vrrp (1-255) shutdown

Переведите маршрутизатор в режим административного завершения работы. VRRP не будет активирован для этого маршрутизатора, пока эта команда не будет удалена вместе с формой no

1.8.7.2 Глобальная конфигурация

Показать команды, глобальные значения по умолчанию и команды настройки отладки

show vrrp [interface INTERFACE] [(1-255)]

Показывает состояние VRRP для некоторых или всех настроенных VRRP-маршрутизаторов. При указании интерфейса будут показаны только маршрутизаторы, настроенные на этом интерфейсе. При указании VRID будут показаны только маршрутизаторы с этим VRID.

debug vrrp [{protocol|autoconfigure|packets|sockets|ndisc|arp|zebra}]

Переключение журналов отладки для компонентов VRRP. Если ни один компонент не указан, отладка для всех компонентов включается / выключается protocol

vrrp default <advertisement-interval

Настройте значения по умолчанию для новых VRRP-маршрутизаторов. Эти значения не повлияют на уже настроенные VRRP-маршрутизаторы, но будут применены к вновь настроенным.

1.9 Настройка правил межсетевого экрана

Управление трафиком осуществляется с помощью утилиты **iptables** (утилита командной строки, являющая стандартным интерфейсом управления работой межсетевого экрана). Межсетевой экран работает по принципу цепочек правил. Для использования утилиты **iptables** необходимо войти в режим Shell:

shell

1.9.1 Правила и действия

Для каждого типа пакетов можно установить набор правил, которые по очереди будут проверяться на соответствие с пакетом и если пакет соответствует, то применять к нему указанное в правиле действие. Правила образуют цепочку, поэтому input, output и forward называют цепочками, цепочками правил. Действий может быть несколько:

- **ACCEPT** - разрешить прохождение пакета дальше по цепочке правил;
- **DROP** - удалить пакет;
- **REJECT** - отклонить пакет, отправителю будет отправлено сообщение, что пакет был отклонен;
- **LOG** - сделать запись о пакете в лог файл;
- **QUEUE** - отправить пакет пользовательскому приложению.

Обрабатываемые пакеты проходят через заданные пользователем цепочки правил до тех пор, пока не будут приняты (**ACCEPT**), отклонены (**REJECT**) или не обработаны (**DROP**).

Правила могут проверять любые соответствия, например, по ip, по порту получателя или отправителя, заголовкам пакета и многому другому. Если пакет не подходит ни одному из правил, то к нему применяется действие по умолчанию, обычно ACCEPT.

Когда мы разобрались с правилами, можно вернутся обратно к цепочкам. Кроме перечисленных выше, есть еще две дополнительные цепочки правил:

- **prerouting** - в эту цепочку пакет попадает перед обработкой iptables, система еще не знает куда он будет отправлен, в input, output или forward;
- **postrouting** - сюда попадают все проходящие пакеты, которые уже прошли цепочку forward.

1.9.2 Таблицы ipatables

Над цепочками правил в iptables есть еще один уровень абстракции, и это таблицы.

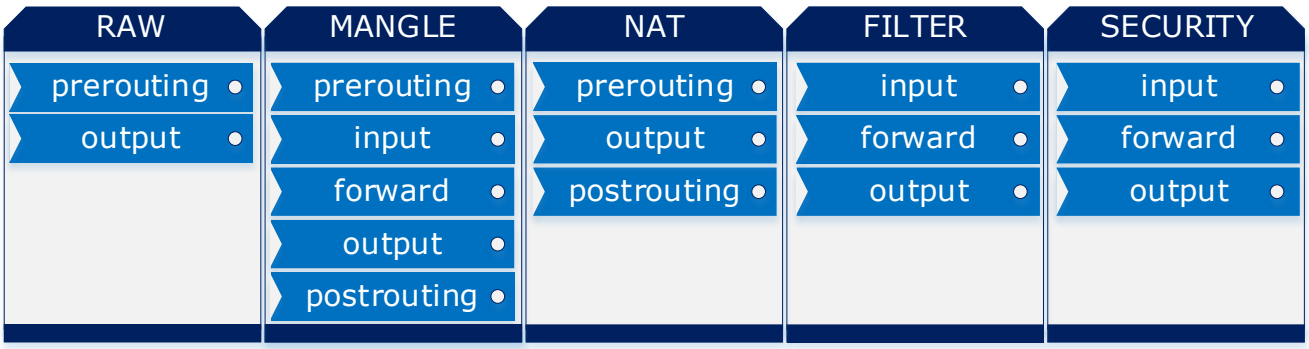


Рисунок 19

В системе есть несколько таблиц, и все они имеют стандартный набор цепочек input, forward и output. Таблицы предназначены для выполнения разных действий над пакетами, например для модификации или фильтрации. Сейчас это для вас не так важно и будет достаточно знать что фильтрация пакетов iptables осуществляется в таблице filter. Но мы рассмотрим их все:

Для упрощения настройки межсетевой экран предоставляет абстрагированный от iptables интерфейс конфигурирования, что является достаточным в большинстве случаев.

Таблица Filter предназначена для фильтрации трафика, то есть разрешения и запрета пакетов и соединений. Для таблицы Filter существуют три базовых цепочки:

INPUT – цепочка для входящих пакетов;

FORWARD – цепочка для перенаправляемых пакетов;

OUTPUT – цепочка для отправляемых пакетов.

Таблица NAT предназначена для операций stateful-преобразования (преобразования на основании данных о соединении) сетевых адресов и портов обрабатываемых пакетов. Для таблицы NAT существуют две базовых цепочки:

PREROUTING – в эту цепочку пакеты попадают до принятия решения о маршрутизации;

POSTROUTING – через эту цепочку проходят все исходящие пакеты;

OUTPUT – через эту цепочку проходят пакеты, сгенерированные процессами хоста.

Таблица Mangle предназначена для операций по классификации и маркировке пакетов и соединений, а также модификации заголовков пакетов (поля TTL и TOS). Для таблицы Mangle существуют следующие базовые цепочки:

PREROUTING — позволяет модифицировать пакет до принятия решения о маршрутизации.

INPUT — позволяет модифицировать пакет, предназначенный самому хосту.

FORWARD — цепочка, позволяющая модифицировать транзитные пакеты.

OUTPUT — позволяет модифицировать пакеты, исходящие от хоста.

POSTROUTING — дает возможность модифицировать все исходящие пакеты, как сгенерированные самим хостом, так и транзитные.

Талица Raw предназначена для выполнения действий с пакетами до их обработки системой conntrack (отслеживание состояний соединений). Для таблицы Raw существуют следующие базовые цепочки:

PREROUTING — в эту цепочку входящие пакеты попадают раньше, чем в любую другую из цепочек iptables, и до обработки их системой conntrack.

OUTPUT — аналогично для пакетов, сгенерированных хостом.

1.9.3 Утилита iptables

Утилита **iptables** предназначена для управления таблицами маршрутизации и NAT.

1.9.3.1 Синтаксис

```
iptables [-t <таблица>] [<опции>]
```

Таблица 20 – Таблицы утилиты iptables

Таблица	Описание
filter	Таблица по умолчанию. Данная таблица содержит предопределённые цепочки INPUT (для входящих), FORWARD (для перенаправляемых пакетов) и OUTPUT (для исходящих пакетов).
nat	Данная таблица используется для пакетов, устанавливающих новое соединение. В ней содержится три предопределённых цепочки: PREROUTING (для изменения входящих пакетов), OUTPUT (для изменения локально сгенерированных пакетов перед их отправлением) и POSTROUTING (для изменения всех исходящих пакетов).
mangle	Данная таблица используется для специальных изменений пакетов. В ней содержатся цепочки PREROUTING (для изменения входящих пакетов до их перенаправления-маршрутизации), OUTPUT (для изменения локально сгенерированных пакетов перед их маршрутизацией), INPUT (для изменения входящих пакетов), FORWARD (для изменения перенаправляемых пакетов) и POSTROUTING (для изменения исходящих пакетов).
raw	Используется преимущественно для создания исключений в слежении за соединениями совместно с целью NOTRACK. Таблица содержит следующие предопределённые цепочки: PREROUTING (для пакетов приходящих из сетевых интерфейсов) OUTPUT (для пакетов генерируемых локальными процессами)

Таблица 21 – Опции команды iptables

Опция	Описание
Основные	
-A	- добавить правило в цепочку;
-C	проверить все правила;
-D	удалить правило;
-I	вставить правило с нужным номером;

Опция	Описание
-L	вывести все правила в текущей цепочке;
-S	вывести все правила;
-F	очистить все правила;
-N	создать цепочку;
-X	удалить цепочку;
-P	установить действие по умолчанию.
Дополнительные	
-p	указать протокол, один из tcp, udp, udplite, icmp, icmpv6, esp, ah, sctp, mh;
-s	указать ip адрес устройства-отправителя пакета;
-d	указать ip адрес получателя;
-i	входной сетевой интерфейс;
-o	исходящий сетевой интерфейс;
-j	выбрать действие, если правило подошло.

1.9.3.1 Пример использования

1.9.3.1.1 Основные действия

Отобразить статус.

```
iptables -L -n -v
```

Отобразить список правил с номерами строк.

```
iptables -n -L -v --line-numbers
```

Отобразить цепочку правил OUTPUT.

```
iptables -L OUTPUT -n -v --line-numbers
```

Удалить все правила.

```
iptables -F
```

Заблокировать все входящие запросы порта 80.

```
iptables -A INPUT -p tcp --dport 80 -j DROP
iptables -A INPUT -i emp -p tcp --dport 80 -j DROP
```

1.9.3.1.2 Список правил

Просмотр правил iptables:

```
iptables -L
```

Просмотр правил с указанием интересующей цепочки:

```
iptables -L INPUT
```

1.9.3.1.3 Очистка правил

Очистка правил:

```
iptables -F
```

Очистка правил только для определенной цепочки:

```
iptables -F Input
```

1.9.3.1.4 Правила по умолчанию

Если для пакета не подходит ни одно правило, то для него применяется действие по умолчанию. Его можно задать с помощью опции -p:

```
iptables -p INPUT ACCEPT
iptables -p OUTPUT ACCEPT
iptables -p FORWARD DROP
```

Разрешить цепочки INPUT и OUTPUT, но запретить FORWARD:

```
iptables -L
```

1.9.3.1.5 Блокировка пакетов

Для блокировки пакетов можно использовать действие DROP. Фильтровать пакеты, которые нужно заблокировать можно по различным критериям, например, протоколу, ip адресу, маске сети, порту и тд.

Пример команды, которая позволяет добавить правило iptables для блокировки всех входящих пакетов от ip 10.10.10.10:

```
iptables -A INPUT -s 10.10.10.10 -j DROP
```

Пример команды, которая позволяет добавить правило iptables для блокировки всех исходящих пакетов на этот же адрес:

```
iptables -A OUTPUT -s 10.10.10.10 -j DROP
```

Пример команды, которая блокирует диапазон ip. Маска сети 10.10.10.0/24. Это все адреса начиная с 10.10.10.0 до 10.10.10.255:

```
iptables -A INPUT -s 10.10.10.0/24 -j DROP
```

Расширенный вариант маски:

```
iptables -A INPUT -s 10.10.10.0/255.255.255.0 -j DROP
```

Заблокировать все входящие соединения ssh:

```
iptables -A INPUT -p tcp --dport ssh -s 10.10.10.10 -j DROP
```

1.9.3.1.6 Удаление правил

Удалние правил iptables выполняется так же, как и создание новых. При этом вместо опции **A** используется опция **D**.

Вывести список правил:

```
iptables -L
```

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  10.10.10.10          anywhere

Chain FORWARD (policy DROP)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
DROP       all  --  10.10.10.10          anywhere
```

Рисунок 20

Удалить правило iptables, которое было создано вторым:

```
iptables -A OUTPUT -s 10.10.10.10 -j DROP
```

Полностью очистить iptables выполнив команду с опцией -F:

```
iptables –F
```

Сохранить настройки iptables можно выполнив команду:

```
iptables-cfg-save
```

Восстановить ранее сохранённые настройки iptables можно выполнив команду:

```
iptables-cfg-restore
```

Просмотреть ранее сохранённые настройки iptables можно выполнив команду:

```
iptables-cfg-show
```

1.10 Настройка функций безопасности

1.10.1 Конфигурирование порта управления

Для обеспечения требований безопасности необходимо ограничить возможность удаленного управления устройством по протоколу SSH. Подключение должно быть разрешено только через специально выделенный порт управления, либо отключено. Настройка порта управления осуществляется при помощи утилиты.

Пример:

```
#Разрешить входящие соединения по протоколу SSH только на интерфейсе eth0
iptables –A INPUT –I eth0 –p TCP --dport 22 -J ACCEPT
…
…
…
iptables –P INPUT DROP
```

1.10.2 Подсистема регистрации событий безопасности.

Подсистема реализована системной службой syslogd, которая осуществляет журналирование событий, происходящих в системе и сообщений ядра системы, а также поддерживает отправку событий на удаленный сервер по протоколу syslog.

Для настройки отправки событий необходимо указать источник, объем событий и адрес централизованной системы мониторинга. Настройки данных параметров хранятся в конфигурационном файле syslog.conf

Конфигурационный файл **syslog.conf** является главным конфигурационным файлом для службы **syslogd** является конфигурационный файл **syslog.conf**. Файл **syslog.conf** представляет собой **набор правил**. Каждое **правило** представляет из себя строку, состоящую из *селектора* и *действия*, разделенных пробелом или табуляцией. **Селектор** представляет собой запись в виде <*источник*>.<*приоритет*>. (*источник* иногда именуют - *категорией*) **Селектор** может состоять из нескольких записей <*источник*>.<*приоритет*>, разделенных символом ";" . Можно указывать несколько источников в одном селекторе (через запятую). Поле **действие** - устанавливает журналируемое действие для селектора.

Сообщения, предназначенные для записи в журнал, проверяются на соответствие шаблонам определяемым селектором. Если соответствует, то выполняется указанное в правиле действие.

Сообщения с уровнем, *равным* или *выше* указанного в селекторе, и источником, *равным указанному* в селекторе, считается *подходящим*. **Звездочка перед** точкой соответствует *любому источнику*, **после** точки - *любому уровню*. Слово **none** после точки - никакому уровню для данного источника. Можно указывать несколько источников в одном селекторе (через запятую).

Источник (категория) может быть следующим:

- 0 - **kern** - Сообщения ядра
- 1 - **user** - Сообщения пользовательских программ
- 2 - **mail** - Сообщения от почтовой системы.
- 3 - **daemon** - Сообщения от тех системных служб, не имеющих категорий.
- 4 - **auth** – Сообщения, связанные с авторизацией пользователей (безопасность/права доступа: login, su и т.д.)
- 5 - **syslog** – Сообщения системы протоколирования.
- 6 - **lpr** - Сообщения от системы печати.
- 7 - **news** - Сообщения от сервера новостей (не используется).
- 8 - **uucp** - Сообщения от UNIX-to-UNIX Copy Protocol (не используется).
- 9 - **cron** - Сообщения от системного планировщика.
- 10 - **authpriv** - Сообщения, связанные с авторизацией пользователей, доступные только определенным пользователям.
- 11 - **ftp** - Сообщения FTP сервера.
- 12 - **NTP** - сообщения сервера времени
- 13 - **log audit**
- 14 - **log alert**
- 15 - **clock daemon -** сообщения службы времени
- с 16 по 23 **local0 - local7** Зарезервированные категории для использования администратором системы. Категория local7 обычно используется для сообщений, генерируемых на этапе загрузки системы.
- **mark** (не имеющая цифрового эквивалента) - присваивается отдельным сообщениям, формируемым самим сервисом syslogd

Приоритет (степени важности) сообщений имеет 8 уровней, которые кодируются числами от 0 до 7:

- 0 - **emerg** (старое название **PANIC**) - Чрезвычайная ситуация. Система неработоспособна.
- 1 - **alert -** Тревога! Требуется немедленное вмешательство.
- 2 - **crit -** Критическая ошибка (критическое состояние).
- 3 - **err** (старое название **ERROR**) - Сообщение об ошибке.
- 4 - **warning** (старое название **WARN**) - Предупреждение.
- 5 - **notice -** Информация о каком-то нормальном, но важном событии.

- 6 - **info -** Информационное сообщение.
- 7 - **debug -** Сообщения, формируемые в процессе отладки.

Согласно **действию,** указанному в правиле, сообщение может быть записано в следующие назначения:

Обычный файл

Задается полным путем, начиная со слеша (/).

Удаленная машина

Для отправки сообщений на другой хост, необходимо перед адресатом добавить символ @.

```
# Пример конфигурационного файла syslogd.
# Все сообщения перенаправляются на
# удалённую сетевую машину.
*.*         @192.168.10.10
```

1.10.3 **Подсистема проверки целостности**

Проверка целостности системных файлов осуществляется автоматически с периодичностью раз в сутки утилитой afick. Ручной контроль целостности осуществляется посредством команды afick с ключом -k.

На этапах ввода системы в опытную, промышленную эксплуатацию необходимо пересоздать базу данных с контрольными суммами файлов и настроить отправку данных на централизованный сервер мониторинга событий безопасности (в случае необходимости).

1.10.3.1 Утилита afick

Afick — утилита, помогающая при обнаружении вторжений, а также позволяющая контролировать общую целостность системы.

Afick контролирует изменения в файловой системе и сразу сообщает вам о них, таким образом, предоставляя вам выбор решить, действительно ли ожидались эти изменения. Эта информация может помочь вам в расследовании инцидента, когда необходимо определить, какие были произведены изменения в системе в результате взлома.

В процессе установки Afick формирует базу данных файлов, каталогов и соответствующих им MD5 контрольных сумм. Файлы и каталоги, включенные в эту базу данных, выбираются соответственно входным данным из файла конфигурации Afick, называемого **afick.conf**, после того, как Afick установит этот файл в /etc каталог. Файл конфигурации afick.conf имеет простую синтаксическую структуру. По вашему усмотрению Вы можете очень быстро добавить или удалить типы файлов, каталоги, и т.д. Ниже приведено содержимое файла afick.conf. Обратите внимание, что элементы в файле конфигурации чувствительны к регистру.

```
# afick config sample file
# directives
#############
database:=/var/lib/afick/afick - Определяет какую базу данных будет использовать Afick
# report_url := stdout - Определяет куда Afick будет выводить результаты своей работы
# verbose := no
# warn_dead_symlinks := no
```

```
# report_full_newdel := no

# warn_missing_file := no

# running_files := no

# timing := no

# text files
```

exclude_suffix := log LOG html htm HTM txt TXT xml - Определяет, что Afick должен игнорировать текстовые файлы с такими расширениями.

```
# help files
```

exclude_suffix := hlp pod chm - Определяет, что Afick должен игнорировать файлы справки с такими расширениями

```
# old files
```

exclude_suffix := tmp old bak - Определяет, что Afick должен игнорировать временные файлы с такими расширениями

```
# fonts
```

exclude_suffix := fon ttf TTF - Определяет, что Afick должен игнорировать файлы шрифтов с такими расширениями

```
# images
```

exclude_suffix := bmp BMP jpg JPG gif png ico - Определяет, что Afick должен игнорировать файлы изображений с такими расширениями

```
# audio
```

exclude_suffix := wav WAV mp3 avi - Определяет, что Afick должен игнорировать медиа файлы с такими расширениями

```
# macros

########

# used by cron
```

@@define MAILTO root - Определяет пользователя, которому будут отсылаться отчеты по работе Afick.

@@define LINES 1000 - Определяет максимальное количество строк в отчете

```
# list the file or directories to scan

# syntaxe :

# file action

# to have action on file (see below

# ! file

# to remove file from scan

# file with blank character have to be quoted
```

 # action : a list of item to check - Ниже описаны опции, определяющие какие атрибуты файлов нужно контролировать.

```
# md5 : md5 checksum

# d : device

# i : inode

# p : permissions
```

```
# n : number of links
# u : user
# g : group
# s : size
# b : number of blocks
# m : mtime
# c : ctime
# a : atime
#R: p+d+i+n+u+g+s+m+c+md5
#L: p+d+i+n+u+g
# action alias may be configured with
# your_alias = another_alias|item[+item][-item]
# all is a pre-defined alias for all items except "a"
# alias :
#########
DIR = p+i+n+u+g
ETC = p+d+i+u+g+s+md5
Logs = p+n+u+
MyRule = p+d+i+n+u+g+s+b+md5+m .
# files to scan
###############
=/ DIR - Проверка с использованием описанных выше правил для каталогов
#
/bin MyRule
/boot MyRule
!/boot/map - Игнорируется указанный каталог.
!/boot/System.map - Игнорируется указанный файл
/etc ETC
/etc/mtab ETC - i
/etc/adjtime ETC - md5
/etc/aliases.db ETC - md5
/etc/mail/statistics ETC - md5
!/etc/map
!/etc/webmin/sysstats/modules/
!/etc/cups/certs/0
/lib MyRule
/lib/modules MyRule -m
```

```
/root MyRule
!/root/.viminfo
!/root/.bash_history
!/root/.mc
/sbin MyRule
/usr/bin MyRule
/usr/sbin MyRule
/usr/lib MyRule
/usr/local/bin MyRule
/usr/local/sbin MyRule
/usr/local/lib MyRule
/var/ftp MyRule
/var/log Logs
/var/www MyRule
```

1.10.3.2 Пример использования

В данном разделе приведен пример, в котором к проверке целостности Afick добавляется основной каталог системы. К примеру, если необходимо, чтобы файлы в основном каталоге проверялись на изменения при монопольном доступе, изменение прав доступа, изменения размера файлов и времени последнего обращения к файлу.

Для начала нужно создать новый элемент, под разделом **#alias** в файле конфигурации afick.conf, как показано ниже:

```
HOME = u+g+p+m+s
```

Затем в разделе **#files to scan** необходимо добавить следующую строку:

```
/home/yourusername HOME
```

Теперь, при следующем запуске, Afick добавит данный каталог в свою базу данных и будет контролировать находящиеся в нем файлы, согласно заданным критериям. Если нужно, чтобы изменения применились немедленно, то можно запустить Afick вручную, используя следующую команду:

```
Afick -- update
```

Иначе придется ждать запуска крон задачи Afick. Эта задача добавляется автоматически во время инсталляции программы и запускается один раз в день. Результаты работы данного задания будет получен по электронной почте на адрес, указанный в разделе **MAILTO** файла конфигурации **afick.conf**. Используя почтового клиента, можно будет увидеть ежедневный отчет приблизительно в следующем виде:

```
This is an automated report generated by Another File Integrity Checker on
+localhost.localdomain at 07:46:07 AM on 02/25/2004.
Output of the daily afick run:
new file : /var/log/afick/afick.log.2
```

```
new file : /var/log/afick/error.log.2
deleted file : /etc/sysconfig/iptables
changed file : /etc/adjtime
changed file : /etc/aliases.db
changed file : /etc/mail/statistics
changed file : /etc/prelink.cache
changed file : /etc/printcap
detailed changes
changed file : /etc/adjtime
MD5 : 7+bTDZQbxsTXEJXhyI2GCw ao6a/yDwoBR8GSL1AKlWXQ
changed file : /etc/aliases.db
MD5 :GT/eP5D+B8apNoa7L5CLRw soh7MnLDuQw4gI9KHlhpTA
changed file : /etc/mail/statistics
MD5 :oshq17jZ2a0o5pYhVBRgwQ vb69gMWXvpIEEZ4fmOl9/Q
changed file : /etc/prelink.cache
MD5 : SKh/403FRMUqBNdCIInQ9A zeC+5EPFfWBR4OeT7xZdbw
changed file : /etc/printcap
MD5 : b5e3g2//bGaxeCxVyRJqaw QFY1NJGy/kdt32B1YV0TXQ
filesize : 194 581
```

В примере выше Afick сообщает, что некоторые файлы были изменены, созданы или удалены. Также показаны начальные и текущие контрольные суммы файлов, и сообщается, что в одном из файлов был изменен его размер. Afick проконтролирует наличие изменений в файле, сравнивая его атрибуты с атрибутами, которые были сохранены при последнем запуске Afick. Примером этого могут служить файлы в папке **/usr/bin** или в **/sbin**. Как правило эти файлы изменяются не часто, если только их не изменили, обновляя программу (в противном случае они не останутся неизменны).

Следует обратить внимание на то, где сохраняется вашу базу данных Afick (по умолчанию — **/var/lib/afick/**), так как возможно возникновение ситуации, когда система была взломана, ну это не было зафиксировано, так как была нарушена целостность базы данных. Возможным решением данного вопроса может быть сохранение базы на защищенных от записи носителях (например, CD-ROM), после чего изменить файл конфигурации afick.conf, чтобы указать на выбранное вами место сохранения базы.

1.10.3.3 Проведение процедуры контроля целостности ОС

Для **сертифицированной редакции** в состав репозитория ОС включена база данных утилиты контроля целостности afick, которая содержит перечень контролируемых бинарных исполняемых файлов изделия, согласно документации на изделие. Для самостоятельной проверки целостности ОС необходимо сделать следующее:

1. Запустить проверку файлов ОС командой:

```
# afick -k
...
```

```
new file : /var/lib/afick/redos/redos.ctr

new file : /var/lib/afick/redos/redos.db

deleted file : /lib/modules/4.19.79-1.el7.x86_64.debug/kernel/arch/x86/crypto/aegis128-aes
ni.ko

...

deleted file : /var/lib/afick/afick.db

        parent_date               : Fri Apr 24 09:17:05 2020

changed file : /etc/afick.conf

        md5                       : 03bf42d0327b3f2fe195d0eca359b1ec       addfc78d9551417
18f78df7587ae3ceb

        filemode                  : 100600        100644

        filesize                  : 924772        924774

# Hash database : 6793 files scanned, 6320 changed (new : 2; delete : 6317; changed : 1; d
angling : 0; exclude_suffix : 0; exclude_prefix : 0; exclude_re : 0; degraded : 3)

# ################################################################

# MD5 hash of /var/lib/afick/redos/redos => oRTAm4SAHdk9vwSktXz88A

# user time : 12.73; system time : 3.14; real time : 27
```

2. После завершения проверки проанализировать полученные данные. Новые файлы можно не принимать во внимание.

```
new : 2
```

Удалённые файлы в отчете можно не учитывать — это объясняется тем, что БД afick снимается для всех контролируемых файлов, а в используемом экземпляре ОС могут использоваться не все пакеты.

```
delete : 6317
```

Интерес в первую очередь представляют изменённые файлы.

```
changed : 1
```

3. Проанализировать подробный отчёт в выводе утилиты выше и оценить, допустимы ли данные изменения файлов.

```
changed file : /etc/afick.conf

        md5                       : 03bf42d0327b3f2fe195d0eca359b1ec       addfc78d9551417
18f78df7587ae3ceb

        filemode                  : 100600        100644

        filesize                  : 924772        924774
```

В приведенном примере был изменен конфигурационный файл утилиты afick. Это считается допустимым изменением, так как на эталонной ОС конфигурация afick по умолчанию не настроена. Скачанный конфигурационный файл, содержащийся в пакете БД afick, заменил файл, поставляемый с утилитой. Других изменений в системе нет, значит, можно сделать вывод о целостности ОС.

1.10.4 Подсистема криптозащиты каналов связи

Устройство поддерживает следующие решения: Infotecs ViPNet VPN, TCC Diamond VPN, OpenVPN, IPSec, PPTP VPN.

Настройка СКЗИ Infotecs ViPNet VPN, TCC Diamond VPN осуществляется согласно инструкции производителя СКЗИ.

1.10.5 Подсистема аудита

В устройстве реализована подсистема аудита, которая позволяет осуществлять аудит:

- запуск и завершение работы системы;
- чтение, запись и изменение прав доступа к файлам;
- инициация сетевых соединений;
- попытки неудачной авторизации в системе;
- изменение сетевых настроек;
- изменение информации о пользователях и группах;
- запуск и остановка приложений;
- выполнение системных вызовов

Подсистема реализована на базе сервиса **auditd**. Просмотр результатов аудита осуществляется утилитами:

- **aureport** - инструмент для генерации итоговых отчетов на основе логов демона аудита;
- **ausearch** - поиск по журналу аудита;
- **auditctl** - инструмент для управления аудитом, предоставляемого Linux ядром.

1.10.5.1 Сервис auditd

1.10.5.1.1 Описание

auditd - это прикладной компонент системы аудита Linux. Он ведёт протокол аудита на диске. Для просмотра протоколов предназначены команды **ausearch** и **aureport.** Команда **auditctl** позволяет настраивать правила аудита. Кроме того, при загрузке загружаются правила из файла */etc/audit.rules*. Некоторые параметры самого демона можно изменить в файле **auditd.conf.**

1.10.5.1.2 Синтаксис

auditd [**-f**] [**-l**] [**-n**]

Таблица 22 – Опции сервиса auditd

Опция	Описание
-f	Не переходить в фоновый режим (для отладки). Сообщения программы будут направляться в стандартный вывод для ошибок (stderr), а не в файл.
-l	Включить следование по символическим ссылкам при поиске конфигурационных файлов.
-n	Не создавать дочерний процесс. Для запуска из inittab

1.10.5.1.3 Сигналы

Таблица 23 – Сигналы сервиса auditd

Опция	Описание
SIGHUP	перезагрузить конфигурацию - загрузить файл конфигурации с диска. Если в файле не окажется синтаксических ошибок, внесенные в него изменения вступят в силу. При этом в протокол будет добавлена запись о событии DAEMON_CONFIG. В противном случае действия службы будут зависеть от параметров space_left_action, admin_space_left_action, disk_full_action, disk_error_action файла auditd.conf.
SIGTERM	прекратить обработку событий аудита и завершить работу, о чём предварительно занести запись в протокол.
SIGUSR1	создать новый файл для протокола, перенумеровав старые файлы или удалив часть из них, в зависимости от параметра max_log_size_action.

1.10.5.1.4 Файлы

/etc/audit/auditd.conf - файл конфигурации демона аудита **/etc/audit/audit.rules** - правила аудита (загружается при запуске службы)

1.10.5.2 Утилита ausearch

Программа **ausearch** является инструментом поиска по журналу аудита. **ausearch** может также принимать данные со стандартного ввода (stdin) до тех пор, пока на входе будут необработанные данные логов. Все условия, указанные в параметрах, объединяются логическим И. К примеру, при указании **-m** и **-ui** в качестве параметров будут показаны события, соответствующие заданному типу и идентификатору пользователя.

1.10.5.2.1 Синтаксис

ausearch [опции]

1.10.5.2.2 Сигналы

Таблица 24 – Сигналы сервиса ausearch

Опция	Описание
-a, **--event** *audit-event-id*	Искать события с заданным *идентификатором события*. Сообщения обычно начинаются примерно так: msg=audit(1116360555.329:2401771). Идентификатор события - это число после ':'. Все события аудита, связанные с одним системным вызовом имеют одинаковый идентификатор.
-c, **--comm** *comm-name*	Искать события с заданным *comm name*. comm name - имя исполняемого файла задачи.
-f, **--file** *file-name*	Искать события с заданным *именем файла*.
-ga, **--gid-all** *all-group-id*	Искать события с заданным эффективным или обычным *идентификатором группы*.

Опция	Описание
-ge, **--gid-effective** *effective-group-id*	Искать события с заданным *эффективным идентификатором группы* или именем группы.
-gi, **--gid** *group-id*	Искать события с заданным *идентификатором группы* или именем группы.
-h, **--help**	Справка
-hn, **--host** *host-name*	Искать события с заданным *именем узла*. Имя узла может быть именем узла, полным доменным именем или цифровым сетевым адресом.
-i, **--interpret**	Транслировать числовые значения в текстовые. Например, идентификатор пользователя будет оттранслирован в имя пользователя. Трансляция выполняется с использованием данных с той машины, где запущен **ausearch**. Т.е. если вы переименовали учетные записи пользователей или не имеете таких же учетных записей на вашей машине, то вы можете получить результаты, вводящие в заблуждение.
-if, **--input** *file-name*	Использовать указанный *файл* вместо логов аудита. Это может быть полезно при анализе логов с другой машины или при анализе частично сохраненных логов.
-k, **--key** *key-string*	Искать события с заданным *ключевым словом*.
-m, **--message** *message-type* | *comma-sep-message-type-list*	Искать события с заданным *типом*. Вы можете указать *список значений, разделенных запятыми*. Можно указать несуществующий в событиях тип **ALL**, который позволяет получить все сообщения системы аудита. Список допустимых типов большой и будет показан, если указать эту опцию без значения. Тип сообщения может быть строкой или числом. В списке значений этого параметра в качестве разделителя используются запятые и пробелы недопустимы.
-o, **--object** *SE-Linux-context-string*	Искать события с заданным *контекстом* (объектом).
-p, **--pid** *process-id*	Искать события с заданным *идентификатором процесса*.
-pp, **--ppid** *parent-process-id*	Искать события с заданным *идентификатором родительского процесса*.
-r, **--raw**	Необработанный вывод. Используется для извлечения записей для дальнейшего анализа.
-sc, **--success** *syscall-name-or-value*	Искать события с заданным *системным вызовом*. Вы можете указать его номер или имя. Если вы указали имя, оно будет проверено на машине, где запущен **ausearch**.
-se, **--context** *SE-Linux-context-string*	Искать события с заданным *контекстом SELinux* (stcontext/subject или *tcontext*/object).
-su, **--subject** *SE-Linux-context-string*	Искать события с заданным контекстом SELinux - *scontext* (subject).

Опция	Описание
-sv, **--success** *success-value*	Искать события с заданным *флагом успешного выполнения*. Допустимые значения: **yes (успешно)** и **no**(неудачно).
-te, **--end** [*end-date*] [*end-time*]	Искать события, которые произошли раньше (или во время) указанной временной точки. Формат даты и времени зависит от ваших региональных настроек. Если дата не указана, то подразумевается текущий день (**today**). Если не указано время, то подразумевается текущий момент (**now**). Используйте 24-часовую нотацию времени, а не AM/PM. Например, дата может быть задана как 10/24/2005, а время - как 18:00:00. Вы можете также использовать ключевые слова: **now** (сейчас), **recent**, **today**, **yesterday**, **this-week**, **this-month**, **this-year**. **today** означает первую секунду после полуночи текущего дня. **recent** - 10 минут назад. **yesterday** - первую секунду после полуночи предыдущего дня. **this-week** означает первую секунду после полуночи первого дня текущей недели, первый день недели определяется из ваших региональных настроек (см. **localtime**). **this-month** означает первую секунду после полуночи первого числа текущего месяца. **this-year** означает первую секунду после полуночи первого числа первого месяца текущего года.
-ts, **--start** [*start-date*] [*start-time*]	Искать события, которые произошли после (или во время) указанной временной точки. Формат даты и времени зависит от ваших региональных настроек. Если дата не указана, то подразумевается текущий день (**today**). Если не указано время, то подразумевается полночь (**midnight**). Используйте 24-часовую нотацию времени, а не AM/PM. Например, дата может быть задана как 10/24/2005, а время - как 18:00:00. Вы можете также использовать ключевые слова: **now** (сейчас), **recent**, **today**, **yesterday**, **this-week**, **this-month**, **this-year**. **today** означает первую секунду после полуночи текущего дня. **recent** - 10 минут назад. **yesterday** - первую секунду после полуночи предыдущего дня. **this-week** означает первую секунду после полуночи первого дня текущей недели, первый день недели определяется из ваших региональных настроек (см. **localtime**). **this-month** означает первую секунду после полуночи первого числа текущего месяца. **this-year** означает первую секунду после полуночи первого числа первого месяца текущего года.
-tm, **--terminal** *terminal*	Искать события с заданным *терминалом*. Некоторые демоны (такие как cron и atd) используют имя демона как имя терминала.
-ua, **--uid-all** *all-user-id*	Искать события, у которых любой из идентификатора пользователя, эффективного идентификатора пользователя или loginuid (auid) совпадают с заданным *идентификатором пользователя*.
-ue, **--uid-effective** *effective-user-id*	Искать события с заданным *эффективным идентификатором пользователя*.

Опция	Описание
-ui, **--uid** *user-id*	Искать события с заданным *идентификатором пользователя*.
-ul, **--loginuid** *login-id*	Искать события с заданным *идентификатором пользователя*. Все программы, которые его используют, должны использовать pam_loginuid.
-v, **--verbose**	Показать версию и выйти
-w, **--word**	Совпадение с полным словом. Поддерживается для имени файла, имени узла, терминала и контекста SELinux.
-x, **--executable** *executable*	Искать события с заданным *именем исполняемой программы*.

1.10.5.3 Утилита aureport

aureport - это инструмент, который генерирует итоговые отчеты на основе логов демона аудита. **aureport** может также принимать данные со стандартного ввода (stdin) до тех пор, пока на входе будут необработанные данные логов. В шапке каждого отчета для каждого столбца есть заголовок - это облегчает понимание данных. Все отчеты, кроме основного итогового отчета, содержат номера событий аудита. Используя их, вы можете найти полные данные о событии с помощью **ausearch -a** *номер события*. Если в отчете слишком много данных, можно задать время начала и время окончания для уточнения временного промежутка. Отчеты, генерируемые **aureport**, могут быть использованы как исходный материал для получения более развернутых отчетов.

1.10.5.3.1 Синтаксис

aureport [*опции*]

1.10.5.3.2 Сигналы

Таблица 25 – Сигналы сервиса aureport

Опция	Описание
-au, **--auth**	Отчет о всех попытках аутентификации
-a, **--avc**	Отчет о всех avc сообщениях
-c, **--config**	Отчет о изменениях конфигурации
-cr, **--crypto**	Отчет о событиях, связанных с шифрованием
-e, **--event**	Отчет о событиях
-f, **--file**	Отчет о файлах
--failed	Для обработки в отчетах выбирать только неудачные события. По умолчанию показываются и удачные и неудачные события.
-h, **--host**	Отчет о хостах
-i, **--interpret**	Транслировать числовые значения в текстовые. Например, идентификатор пользователя будет оттранслирован в имя пользователя. Трансляция выполняется с использованием данных с той машины, где запущен **aureport**. Т.е. если вы переименовали учетные записи пользователей или не имеете таких же учетных

Опция	Описание
	записей на вашей машине, то вы можете получить результаты, вводящие в заблуждение.
-if, **--input** *файл*	Использовать указанный *файл* вместо логов аудита. Это может быть полезно при анализе логов с другой машины или при анализе частично сохраненных логов.
-l, **--login**	Отчет о попытках входа в систему
-m, **--mods**	Отчет об изменениях пользовательских учетных записей.
-ma, **--mac**	Отчет о событиях в системе обеспечивающей мандатное управление доступом - Mandatory Access Control (MAC).
-p, **--pid**	Отчет о процессах
-r, **--response**	Отчет о реакциях на аномальные события
-s, **--syscall**	Отчеты о системных вызовах
--success	Для обработки в отчетах выбирать только удачные события. По умолчанию показываются и удачные и неудачные события.
--summary	Генерировать итоговый отчет, который дает информацию только о количестве элементов в том или ином отчете. Такой режим есть не у всех отчетов.
-t, **--log**	Этот параметр генерирует отчет о временных рамках каждого отчета.
-te, **--end** [*дата*] [*время*]	Искать события, которые произошли раньше (или во время) указанной временной точки. Формат даты и времени зависит от ваших региональных настроек. Если дата не указана, то подразумевается текущий день (**today**). Если не указано время, то подразумевается текущий момент (**now**). Используйте 24-часовую нотацию времени, а не AM/PM. Например, дата может быть задана как 10/24/2005, а время - как 18:00:00. Вы можете также использовать ключевые слова: **now** (сейчас), **recent**, **today**, **yesterday**, **this-week**, **this-month**, **this-year**. **today** означает первую секунду после полуночи текущего дня. **recent** - 10 минут назад. **yesterday** - первую секунду после полуночи предыдущего дня. **this-week** означает первую секунду после полуночи первого дня текущей недели, первый день недели определяется из ваших региональных настроек (см. **localtime**). **this-month** означает первую секунду после полуночи первого числа текущего месяца. **this-year** означает первую секунду после полуночи первого числа первого месяца текущего года.
-tm, **--terminal**	Отчет о терминалах
-ts, **--start** [*дата*] [*время*]	Искать события, которые произошли после (или во время) указанной временной точки. Формат даты и времени зависит от ваших региональных настроек. Если дата не указана, то подразумевается текущий день (**today**). Если не указано время, то подразумевается полночь (**midnight**). Используйте 24-часовую

Опция	Описание
	нотацию времени, а не AM/PM. Например, дата может быть задана как 10/24/2005, а время - как 18:00:00. Вы можете также использовать ключевые слова: **now** (сейчас), **recent**, **today**, **yesterday**, **this-week**, **this-month**, **this-year**. **today** означает первую секунду после полуночи текущего дня. **recent** - 10 минут назад. **yesterday** - первую секунду после полуночи предыдущего дня. **this-week** означает первую секунду после полуночи первого дня текущей недели, первый день недели определяется из ваших региональных настроек (см. **localtime**). **this-month** означает первую секунду после полуночи первого числа текущего месяца. **this-year** означает первую секунду после полуночи первого числа первого месяца текущего года.
-u, **--user**	Отчет о пользователях
-v, **--version**	Вывести версию программы и выйти
-x, **--executable**	Отчет о исполняемых объектах

1.10.5.4 Утилита auditctl

auditctl используется для контроля поведения, получения состояния и добавления/удаления правил аудита, предоставляемого Linux ядром версии 2.6.

1.10.5.4.1 Синтаксис

auditctl [*опции*]

1.10.5.4.2 Сигналы

Таблица 26 – Сигналы сервиса auditctl

Опция	Описание
-b *backlog*	Установить максимальное количество доступных для аудита буферов, ожидающих обработки (значение в ядре по умолчанию - 64). Если все буфера заняты, то флаг сбоя будет выставлен ядром для его дальнейшей обработки.
-e [**0**..**2**]	Установить флаг блокировки. **0** позволит на время отключить аудит, включить его обратно можно, передав **1** как параметр. Если установлено значение опции **2**, то защитить конфигурацию аудита от изменений. Каждый, кто захочет воспользоваться этой возможностью, может поставить эту команду последней в audit.rules. После этой команды все попытки изменить конфигурацию будут отвергнуты с уведомлением в журналах аудита. В этом случае, чтобы задействовать новую конфигурацию аудита, необходимо перезагрузить систему аудита.
-f [**0**..**2**]	Установить способ обработки для флага сбоя. **0**=silent **1**=printk **2**=panic. Эта опция позволяет определить каким образом ядро будет обрабатывать критические ошибки. Например, флаг сбоя выставляется при следующих условиях: ошибки передачи в

Опция	Описание
	пространство демона аудита, превышение лимита буферов, ожидающих обработки, выход за пределы памяти ядра, превышение лимита скорости выдачи сообщений. Значение по умолчанию: **1**. Для систем с повышенными требованиями к безопасности, значение **2** может быть более предпочтительно.
-h	Краткая помощь по аргументам командной строки.
-i	Игнорировать ошибки при чтении правил из файла.
-l	Вывести список всех правил по одному правилу в строке.
-k *ключ*	Установить на правило ключ фильтрации. Ключ фильтрации - это произвольная текстовая строка длиной не больше 31 символа. Ключ помогает уникально идентифицировать записи, генерируемые в хода аудита за точкой наблюдения.
-m *текст*	Послать в систему аудита пользовательское сообщение. Это может быть сделано только из-под учетной записи root.
-p [**r**|**w**|**x**|**a**]	Установить фильтр прав доступа для точки наблюдения. **r**=чтение, **w**=запись, **x**=исполнение, **a**=изменение атрибута. Не путайте эти права доступа с обычными правами доступа к файлу - они определяют типы системных вызовов, которые выполняют данные действия. Заметьте, системные вызовы read и write не включены в этот набор, поскольку логи аудита были бы перегружены информацией о работе этих вызовов.
-r *частота*	Установить ограничение скорости выдачи сообщений в секунду (**0** - нет ограничения). Если эта *частота* не нулевая и она превышается в ходе аудита, флаг сбоя выставляется ядром для выполнения соответствующего действия. Значение по умолчанию: 0.
-R *файл*	Читать правила из *файла*. Правила должны быть расположены по одному в строке и в том порядке, в каком они должны исполняться. Следующие ограничения накладываются на файл: владельцем должен быть root и доступ на чтение должен быть только у него. Файл может содержать комментарии, начинающиеся с символа '#'. Правила, расположенные в файле, идентичны тем, что набираются в командной строке, без указания 'auditctl'.
-s	Получить статус аудита.
-a *список*,*действие*	Добавить правило с указанным *действием* к концу *списка*. Заметьте, что запятая разделяет эти два значения. Отсутствие запятой вызовет ошибку. Ниже описаны имена доступных списков:
task	Добавить правило к списку, отвечающему за процессы. Этот список правил используется только во время создания процесса - когда родительский процесс вызывает fork() или clone(). При

Опция	Описание
	использовании этого списка вы можете использовать только те поля, которые известны во время создания процесса: uid, gid и т.д.
entry	Добавить правило к списку, отвечающему за точки входа системных вызовов. Этот список применяется когда необходимо создать событие для аудита, привязанное к точкам входа системных вызовов.
exit	Добавить правило к списку, отвечающему за точки выхода из системных вызовов. Этот список применяется когда необходимо создать событие для аудита, привязанное к точкам выхода из системных вызовов.
user	Добавить правило, отвечающего за список фильтрации пользовательских сообщений. Этот список используется ядром, чтобы отфильтровать события приходящие из пользовательского пространства, перед тем как они будут переданы демону аудита. Необходимо отметить, что только следующие поля могут быть использованы: uid, auid, gid и pid. Все остальные поля будут обработаны, как если бы они не совпали.
exclude	Добавить правило к списку, отвечающего за фильтрацию событий определенного типа. Этот список используется, чтобы отфильтровывать ненужные события. Например, если вы не хотите видеть avc сообщения, вы должны использовать этот список. Тип сообщения задается в поле msgtype. Ниже описаны доступные *действия* для правил:
never	Аудит не будет генерировать никаких записей. Это может быть использовано для подавления генерации событий. Обычно необходимо подавлять генерацию в верху списка, а не внизу, т.к. событие инициируется на первом совпавшем правиле.
always	Установить контекст аудита. Всегда заполнять его во время входа в системный вызов, и всегда генерировать запись во время выхода из системного вызова.
-A *список*,*действие*	Добавить правило с указанным *действием* в начало *списка*.
-d *список*,*действие*	Удалить правило с указанным *действием* из *списка*. Правило удаляется только в том случае, если полностью совпали и имя системного вызова и поля сравнения.
-D	Удалить все правила и точки наблюдения.
-S [*Имя или номер системного вызова*|**all**]	Любой *номер* или *имя* системного вызова может быть использован. Также возможно использование ключевого слова *all*. Если какой-либо процесс выполняет указанный системный вызов, то аудит генерирует соответствующую запись. Если значения полей сравнения заданы, а системный вызов не указан, правило будет применяться ко всем системным вызовам. В одном правиле может быть задано несколько системных вызовов - это положительно сказывается на производительности, поскольку заменяет обработку нескольких правил.

Опция	Описание
-F [*n*=*v* | *n***!**=*v* | *n*<*v* | *n*>*v* | *n*<=*v* | *n*>=*v* | *n***&**=*v* | *n***&=***v*]	Задать поле сравнения для правила. Атрибуты поля следующие: объект, операция, значение. Вы можете задать до 64 полей сравнения в одной команде. Каждое новое поле должно начинаться с **-F**. Аудит будет генерировать запись, если произошло совпадение по всем полями сравнения. Допустимо использование одного из следующих 8 операторов: равно, не равно, меньше, больше, меньше либо равно, больше либо равно, битовая маска (*n***&***v*) и битовая проверка (*n***&=***v*). Битовая проверка выполняет операцию 'and' над значениями и проверяет, равны ли они. Битовая маска просто выполняет операцию 'and'. Поля, оперирующие с идентификатором пользователя, могут также работать с именем пользователя - программа автоматически получит идентификатор пользователя из его имени. То же самое можно сказать и про имя группы. Поля сравнения могут быть заданы для следующих объектов:
a0, a1, a2, a3	Четыре первых аргумента, переданных системному вызову. Строковые аргументы не поддерживаются. Это связано с тем, что ядро должно получать указатель на строку, а проверка поля по значению адреса указателя не желательна. Таким образом, вы должны использовать только цифровые значения.
arch	Архитектура процессора, на котором выполняется системный вызов. Используйте 'uname -m', чтобы определить архитектуру. Если вы не знаете архитектуру вашей машины, но хотите использовать таблицу 32-х битных системных вызовов, и ваша машина поддерживает 32 бита, вы можете использовать **b32**. Подобно этому **b64** может быть использовано для использования таблицы 64-х битных системных вызовов.
auid	Это аббревиатура: audit uid - идентификатор пользователя, использованный для входа в систему.
devmajor	Главный номер устройства (Device Major Number)
devminor	Вспомогательный номер устройства (Device Minor Number)
egid	Действительный идентификатор группы
euid	Действительный идентификатор пользователя
exit	Значение, возвращаемое системным вызовом при выходе.
fsgid	Идентификатор группы, применяемый к файловой системе.
fsuid	Идентификатор пользователя, применяемый к файловой системе.
gid	Идентификатор группы
Идентификатор группы	**inode**
inode	Номер inode
key	Альтернативный способ установить ключ фильтрации. Смотри выше описание опции **-k**.

Опция	Описание
msgtype	Используется для проверки совпадения с числом, описывающим тип сообщения. Может быть использован только в списке **exclude**.
obj_user	Имя пользователя-владельца ресурса (в контексте SELinux)
obj_role	Роль ресурса (в контексте SELinux)
obj_type	Тип ресурса (в контексте SELinux)
obj_lev_low	Нижний уровень ресурса (в контексте SELinux)
obj_lev_high	Верхний уровень ресурса (в контексте SELinux)
path	Полный путь к файлу для точки наблюдения. Смотри ниже описание опции "**-w**". Может быть использован только в списке **exit**.
perm	Фильтр прав доступа для файловых операций. Смотри выше описание опции "**-p**". Может быть использован только в списке **exit**.
pers	Персональный номер операционной системы.
pid	Идентификатор процесса
ppid	Идентификатор родительского процесса.
subj_user	Имя пользователя-владельца процесса (в контексте SELinux)
subj_role	Роль процесса (в контексте SELinux)
subj_type	Тип процесса (в контексте SELinux)
subj_sen	Чувствительность процесса (в контексте SELinux)
subj_clr	Допуск процесса (в контексте SELinux)
sgid	Установленный идентификатор группы
success	Если значение, возвращаемое системным вызовом, больше либо равно 0, данный объект будет равен "true/yes", иначе "false/no". При создании правила используйте 1 вместо "true/yes" и 0 вместо "false/no".
suid	Установленный идентификатор пользователя
uid	Идентификатор пользователя
-w *путь*	Добавить точку наблюдения за файловым объектом, находящемуся по указанному *пути*. Вы не можете добавлять точку наблюдения к каталогу верхнего уровня - это запрещено ядром. Групповые символы (wildcards) также не могут быть использованы, попытки их использования будут генерировать предупреждающее сообщение. Внутренне точки наблюдения реализованы как слежение за inode. Таким образом, если вы установите точку наблюдения за каталогом, вы увидите файловые события, которые в действительности будут означать обновления метаданных этой inode, и вы можете не увидеть событий, непосредственно связанных с файлами. Если вам необходимо следить за всеми файлами в каталоге, рекомендуется создавать

Опция	Описание
	индивидуальную точку наблюдения для каждого файла. В противоположность к правилам аудита системных вызовов, точки наблюдения не оказывают влияния на производительность, связанную с количеством правил посылаемых в ядро.
-W *путь*	Удалить точку наблюдения за файловым объектом, находящемуся по указанному *пути*.

1.10.5.4.3 Примеры использования для контроля поведения, получения состояния

Чтобы увидеть все системные вызовы, используемые определенным процессом:

auditctl -a entry,always -S all -F pid=1005

Чтобы увидеть все файлы, открытые определенным пользователем:

auditctl -a exit,always -S open -F auid=510

Чтобы увидеть неудачные попытки вызова системной функции 'open':

auditctl -a exit,always -S open -F success!=0

1.10.5.4.4 Создание правил

Список опций команды auditctl для создания правил (подробное описание опций приведено в таблице **28**):

- **-l** — вывести список имеющихся правил;
- **-a** — добавить новое правило;
- **-d** — удалить правило из списка;
- **-D** — удалить все имеющиеся правила.

Чтобы создать новое правило, нужно выполнить команду вида:

$ auditctl -a <список>, <действие> -S <имя системного вызова> -F <фильтры>

Сначала после опции -a указывается список, в который нужно добавить правило. Всего существует 5 таких списков:

- task — события, связанные с созданием новых процессов;
- entry — события, которые имеют место при входе в системный вызов;
- exit — события, которые имеют место при выходе из системного вызова;
- user — события, использующие параметры пользовательского пространства;
- exclude — используется для исключения событий.

Затем указывается, что нужно делать после наступления события. Здесь возможны два варианта: always (события будут записываться в журнал) и never (не будут).

После опции -S идёт имя системного вызова, при котором событие нужно перехватить (open, close и т.п.).

После опции -F указываются дополнительные параметры фильтрации. Например, если нам требуется вести аудит обращений к файлам из каталога /etc, правило будет выглядеть так:

$ auditctl -a exit,always -S open -F path =/etc/

Можно установить и дополнительный фильтр:

$ auditctl -a exit,always -S open -F path =/etc/ -F perm = aw

Аббревиатура aw означает следующее: a — изменение атрибута (attribute change), w — запись (write). Формулировка perm = aw указывает, что для директории /etc нужно отслеживать все факты изменения атрибутов (a — attribute change) и w (w — write).

При настройке слежения за отдельными файлами можно опустить опцию -S, например:

$ auditctl -a exit,always -F path =/etc/ -F perm = aw

1.10.5.5 Файлы правил

Правила можно не только задавать через командную строку, но и прописывать в файле etc/audit/audit.rules.
Начинается этот файл с так называемых метаправил, в которых задаются общие настройки журналирования:

```
# удаляем все ранее созданные правила
-D

# задаём количество буферов, в которых будут храниться сообщения
-b 320

# указываем, что делать в критической ситуации (например, при переполнении буферов): 0 - ничего не делать; 1 - отправлять сообщение в dmesg, 2 -  отправлять ядро в панику
-f 1
```

Далее следуют пользовательские правила. Их синтаксис предельно прост: достаточно просто перечислить соответствующие опции команды auditctl. Рассмотрим пример типового конфигурационного файла:

```
# отслеживать системные вызовы unlink () и rmdir()
-a exit,always -S unlink -S rmdir

# отслеживать системные вызовы open () от пользователя с UID 1001
-a exit,always -S open -F loginuid=1001
```

```
# отслеживать доступ к файлам паролей и групп и попытки их изменения:
-w /etc/group -p wa
-w /etc/passwd -p wa
-w /etc/shadow -p wa
-w /etc/ers -p wa

# отслеживать доступ к следующей директории:
-w /etc/my_directory -p r

# закрыть доступ к конфигурационному файлу для предотвращения изменений
-e 2
```

Изменения конфигурации вступят в силу после перезапуска демона auditd:

```
$  service auditd restart
```

1.11 Работа в режиме RedBox

Маршрутизаторы TOPAZ FW MX240 и MX681 могут быть использованы в качестве устройств RedBox.

Для устройств TOPAZ FW MX240-Bx-C-RB предусматривается следующая предустановленная конфигурация:

- Eth0 – порт A;
- Eth1 – порт B;
- E1p1 – порт interlink;
- E1p2 – порт interlink;
- IP-адрес устройства 192.168.3.127/24.

Для устройств TOPAZ FW MX681-Bx-C-RB предусматривается следующая предустановленная конфигурация:

- Eth0 – порт A;
- Eth1 – порт B;
- Eth2 – порт interlink;
- IP-адрес устройства 192.168.3.127/24.

Примечание Порты Interlink не поддерживают STP/RSTP протоколы и не пересылают BPDU пакеты, использование избыточных соединений приведет к широковещательному шторму и нарушению работы сети. Используйте interlink порты для подключения оконечных SAN устройств или сетей без кольцевой топологии.

2 МАРКИРОВКА И ПЛОМБИРОВАНИЕ

Вся обязательная информация по маркировке нанесена на лицевой и боковой панели. Маркировка выполнена способом, обеспечивающим ее сохранность на все время эксплуатации устройства.

Перечень информации, содержащейся в маркировке на лицевой панели:

- наименование и условное обозначение;
- назначение светодиодов устройства;
- назначение клеммных соединений и разъемов устройства.

Перечень информации, содержащейся в маркировке на боковой панели:

- наименование и условное обозначение;
- товарный знак;

- порядковый номер по системе нумерации предприятия-изготовителя;
- дата изготовления;

Для предотвращения несанкционированного доступа к внутренним электрическим элементам корпус устройства должен быть опломбирован путем нанесения саморазрушающейся наклейки.

3 УПАКОВКА

Устройства размещается в коробке из гофрированного картона.

Эксплуатационная документация уложена в потребительскую тару вместе с устройством.

В потребительскую тару вложена товаросопроводительная документация, в том числе упаковочный лист, содержащий следующие сведения:

- наименование и условное обозначение;
- дату упаковки;
- подпись лица, ответственного за упаковку.

4 ТЕХНИЧЕСКОЕ ОБСЛУЖИВАНИЕ

Техническое обслуживание устройства заключается в профилактических осмотрах.

При профилактическом осмотре должны быть выполнены следующие работы:

- проверка обрыва или повреждения изоляции проводов и кабелей;
- проверка надежности присоединения проводов и кабелей;
- проверка отсутствия видимых механических повреждений, а также пыли и грязи на корпусе устройства.

Периодичность профилактических осмотров устройства устанавливается потребителем, но не реже 1 раз в год.

Эксплуатация устройства с повреждениями категорически запрещается.

5 ТРАНСПОРТИРОВАНИЕ И ХРАНЕНИЕ

Транспортирование устройств должно производиться в упаковке предприятия-изготовителя любым видом транспорта, защищающим от влияний окружающей среды, в том числе авиационным в отапливаемых герметизированных отсеках самолетов.

Размещение и крепление в транспортных средствах упакованных устройств должно обеспечивать его устойчивое положение, исключать возможность ударов друг о друга, а также о стенки транспортных средств.

Укладывать упакованные устройства в штабели следует с правилами и нормами, действующими на соответствующем виде транспорта, чтобы не допускать деформации транспортной тары при возможных механических перегрузках.

При погрузке и выгрузке запрещается бросать и кантовать устройства.

После продолжительного транспортирования при отрицательных температурах приступать к вскрытию упаковки не ранее 12 часов после размещения устройств в отапливаемом помещении.

Устройства следует хранить в невскрытой упаковке предприятия-изготовителя на стеллаже в сухом отапливаемом и вентилируемом помещении, при этом в атмосфере помещения должны отсутствовать пары агрессивных жидкостей и агрессивные газы.

Средний срок сохранности в потребительской таре в отапливаемом помещении, без консервации - не менее 2 лет.

нормальные климатические факторы хранения:

- температура хранения +20 ± 5 ^{0}C;
- значение относительной влажности воздуха: 30-80 %.

Предельные климатические факторы хранения:

- температура хранения от -40 до +70 ^{0}C;
- значение относительной влажности воздуха: верхнее 100% при 30^{0}C.

6 УТИЛИЗАЦИЯ

Устройства не представляют опасности для жизни, здоровья людей и окружающей среды.

Устройства не содержат драгоценных и редкоземельных металлов.

После окончания срока службы, специальных мер по подготовке и отправке устройств на утилизацию не предусматривается.

7 ИСПОЛЬЗОВАНИЕ ПО НАЗНАЧЕНИЮ

7.1 Эксплуатационные ограничения и меры безопасности

К эксплуатации устройства должны допускаться лица, изучившие настоящее руководство по эксплуатации и обладающие базовыми знаниями в области средств вычислительной техники.

Устройство может размещаться вне взрывоопасных зон как на открытом воздухе, так и в помещении. При этом устройство должен быть защищен от прямого воздействия атмосферных осадков. Рабочее положение – вдоль DIN-рейки.

Для нормального охлаждения устройства, а также для удобства монтажа и обслуживания, при монтаже устройства сверху и снизу необходимо предусмотреть свободное пространство не менее 100 мм. Принудительная вентиляция не требуется.

- Производитель не несет ответственность за ущерб, вызванный неправильным монтажом, нарушением правил эксплуатации или использованием оборудования не по назначению.
- Во время монтажа, эксплуатации и технического обслуживания оборудования необходимо соблюдать «Правила технической эксплуатации электроустановок потребителей».
- Монтаж и эксплуатацию оборудования должен проводить квалифицированный персонал, имеющий группу по электробезопасности не ниже 3 и аттестованный в установленном порядке на право проведения работ в электроустановках потребителей до 1000 В.
- На лице, проводящем монтаж, лежит ответственность за производство работ в соответствии с настоящим руководством, требованиями безопасности и электромагнитной совместимости.
- В случае возникновения неисправности необходимо отключить питание от устройства, демонтировать и передать его в ремонт производителю.

7.2 Монтаж

7.2.1 Подготовка к монтажу

Распаковывание устройства следует производить после выдержки упаковки в нормальных условиях не менее двух часов.

При распаковывании следует соблюдать следующий порядок операций:

- открыть коробку;
- из коробки извлечь:
  - вкладыш;
  - комплект монтажный;
  - устройство.

- произвести внешний осмотр устройства:
  - проверить отсутствие видимых внешних повреждений корпуса и внешних разъемов;
  - внутри устройства не должно быть незакрепленных предметов;
  - изоляция не должна иметь трещин, обугливания и других повреждений;
  - маркировка устройства, комплектующих изделий должна легко читаться и не иметь повреждений.

7.2.2 Установка на DIN-рейку

Устройство устанавливается в стойку 19” (монтажный кронштейн высотой 3U) или на монтажную рейку (DIN-профиль 35 мм) в следующей последовательности:

- корпус устройства ставится на рейку, цепляясь верхними выступами;
- корпус опускается вниз относительно верхнего выступа до щелчка.

ВНИМАНИЕ! МОНТАЖНАЯ РЕЙКА (МОНТАЖНЫЙ КРОНШТЕЙН) ДОЛЖНА БЫТЬ ЗАЗЕМЛЕНА.

7.2.3 Внешние подключения

Внешние подключения осуществляются с помощью разъемов MSTBT 2,5/4-ST проводами сечением до 1,5 мм^2.

Рисунок 21 – Внешний вид разъема MSTBT 2,5/4-ST

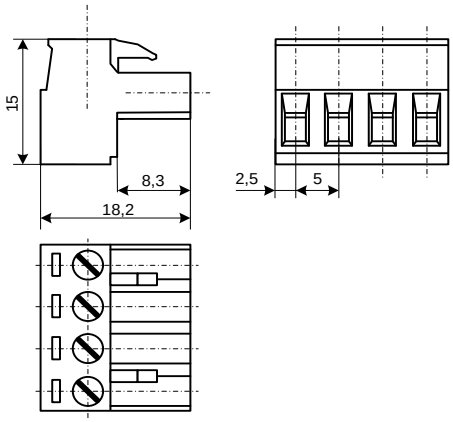


Рисунок 22 – Габаритные размеры разъема MSTBT 2,5/4-ST

ВНИМАНИЕ! ПОДКЛЮЧЕНИЕ К КЛЕММАМ УСТРОЙСТВА ПРОИЗВОДИТЬ ПРИ ОБЕСТОЧЕННОМ ОБОРУДОВАНИИ

ВНИМАНИЕ! ПРИ ПРОВЕРКЕ ГОТОВНОСТИ К РАБОТЕ ПРОВЕРИТЬ ПРАВИЛЬНОСТЬ ПОДКЛЮЧЕНИЙ, КРЕПЛЕНИЕ КЛЕММНИКОВ.

7.2.4 Шина T-BUS

Шина T-BUS представляет собой 5-ти проводную шину, составляемую из произвольного количества единичных Т-образных шинных соединителей ME 22,5 T-BUS 1,5/5-ST-3,81, крепящихся к DIN-рейке с помощью защелок.

Шина T-BUS предназначена для обеспечения питания установленных на ней устройств TOPAZ. Установленные на шине T-BUS устройства, поддерживающие передачу данных по интерфейсу RS-485, также объединяются в единую линию связи RS-485 типа «общая шина».

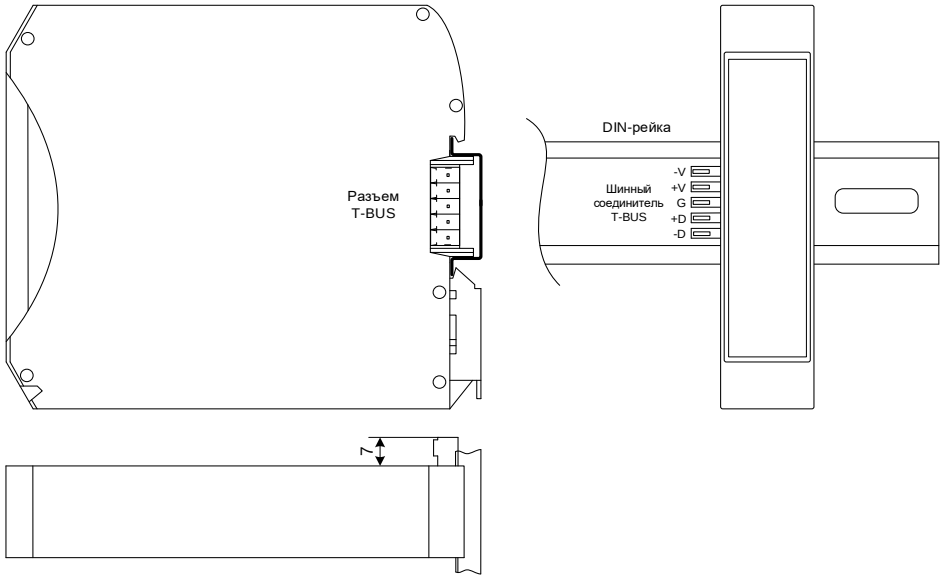


Рисунок 23 – Размещение устройства на DIN-рейке с шиной T-BUS

ВНИМАНИЕ! ПРИ УСТАНОВКЕ УСТРОЙСТВА НА ШИНУ T-BUS НЕОБХОДИМО КОНТРОЛИРОВАТЬ ПОЛОЖЕНИЕ КЛЕММ ШИННОГО СОЕДИНИТЕЛЯ T-BUS ОТНОСИТЕЛЬНО РАЗЪЕМА T-BUS НА ТЫЛЬНОЙ СТОРОНЕ КОРПУСА.

Для подключения к шине T-BUS монтажных проводов используются штекеры MC 1,5/5 ST 3,81 и IMC 1,5/5 ST 3,81. На рисунке ниже приведен внешний вид шиты T-BUS в сборе, где:

A – шинный соединитель ME 22,5 T-BUS 1,5/5-ST-3,81
B – штекер MC 1,5/5-ST-3,81
C – штекер IMC 1,5/5-ST-3,81

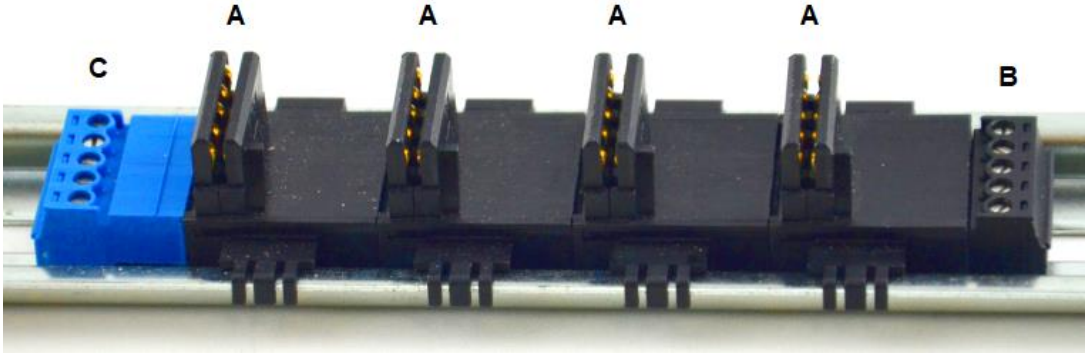


Рисунок 24 – Внешний вид шины T-BUS

Примечание Штекер IMC 1,5/5-ST-3,81 не входит в стандартный комплект поставки устройства.

7.2.5 Подключение питания

Количество и тип каналов питания устройства зависят от исполнения по питанию, согласно заказной кодировке. При наличии напряжения питания на канале питания загорится индикатор **PWR**.

При подключении источника питания постоянного тока к каналу питания 220 В, полярность значения не имеет.

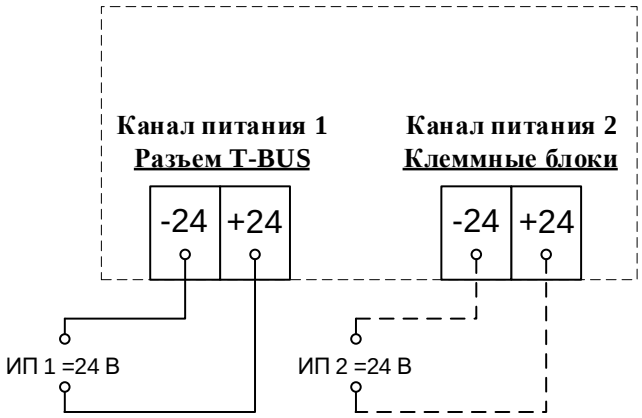


Рисунок 25 – Схема подключения питания

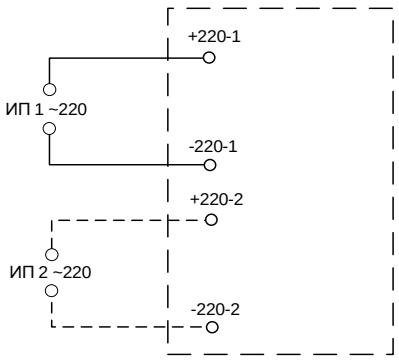


Рисунок 26 – Схема подключения питания устройств исполнения HV (2HV)

ВНИМАНИЕ! ОДНОВРЕМЕННОЕ ПОДКЛЮЧЕНИЕ К СЕТИ ПИТАНИЯ 24 В И 220 В НЕ ПОДДЕРЖИВАЕТСЯ.

ВНИМАНИЕ! СЕТЬ ПИТАНИЯ (≈/= 220 В) ДОЛЖНА ИМЕТЬ ПРОВОД ЗАЗЕМЛЕНИЯ.

7.2.5.1 Питание шины T-BUS

Рекомендуемое напряжение питания шины T-BUS 24 В. Подача питания на шину T-BUS осуществляется одним из следующих способов:

- от внешнего источника питания, подключенного к шине с помощью штекера;
- от источника питания TOPAZ, установленного на шине.

ВНИМАНИЕ! НЕОБХОДИМО УЧИТЫВАТЬ, ЧТОБЫ НОМИНАЛЬНОЕ ЗНАЧЕНИЕ НАПРЯЖЕНИЯ ПИТАНИЯ ШИНЫ T-BUS ВХОДИЛО В ДОПУСТИМЫЙ ДИАПАЗОН ПИТАНИЯ ДЛЯ КАЖДОГО УСТРОЙСТВА TOPAZ, УСТАНОВЛЕННОГО НА ШИНЕ. НОМИНАЛЬНЫЕ ЗНАЧЕНИЯ И ДОПУСТИМЫЕ ДИАПАЗОНЫ ПИТАНИЯ УСТРОЙСТВ TOPAZ ПРИВЕДЕНЫ В РУКОВОДСТВАХ ПО ЭКСПЛУАТАЦИИ НА СООТВЕТСТВУЮЩИЕ УСТРОЙСТВА.

ВНИМАНИЕ! НЕДОПУСТИМО ПОДАВАТЬ ВНЕШНЕЕ НАПРЯЖЕНИЕ ПИТАНИЯ 110/220 В НА ШИНУ T-BUS, ТАК КАК ЭТО ПРИВЕДЕТ К ВЫХОДУ ИЗ СТРОЯ ПОДКЛЮЧЕННЫХ К НЕЙ УСТРОЙСТВ.

7.2.6 Монтаж устройства TOPAZ FW MX710

7.2.6.1 Установка в стойку 19”

Изделие устанавливается в стойку 19” (монтажный кронштейн высотой 2U).

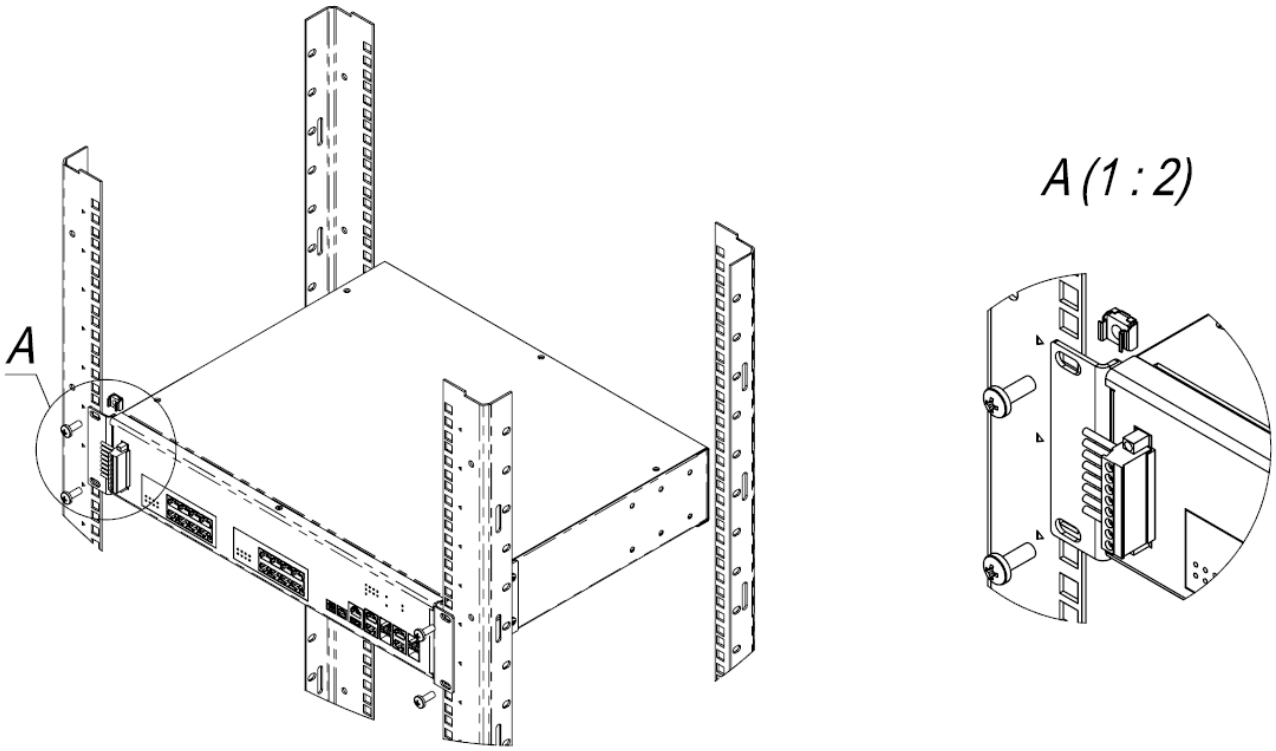


Рисунок 27 - Размещение устройства в стойку 19”

7.2.6.2 Подключение питания

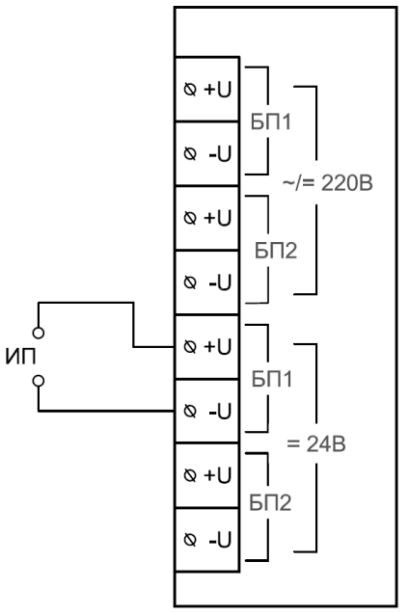
Входы питания модификации TOPAZ FW MX710 располагаются на клеммном блоке. В зависимости от исполнения, устройство может иметь следующие входы питания, каждый из которых обозначен соответствующей маркировкой:

- вход питания для первого блока питания (БП1), если на его вход требуется подавать 24 В постоянного тока;
- вход питания для второго блока питания (БП2), если на его вход требуется подавать 24 В постоянного тока;
- вход питания для первого блока питания (БП1), если на его вход требуется подавать 48 В постоянного тока;
- вход питания для второго блока питания (БП2), если на его вход требуется подавать 48 В постоянного тока;
- вход питания для первого блока питания (БП1), если на его вход требуется подавать 220 В постоянного или переменного тока;
- вход питания для второго блока питания (БП2), если на его вход требуется подавать 220 В постоянного или переменного тока.

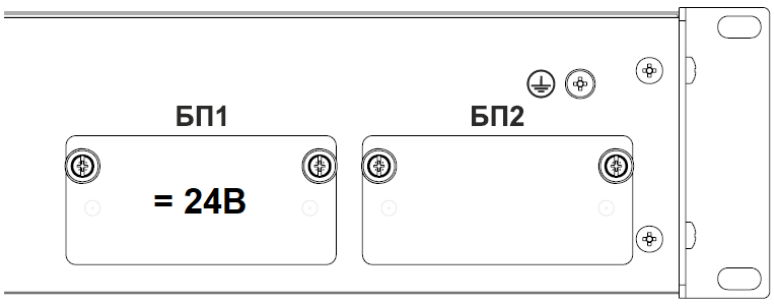
Напряжение, на которое рассчитан каждый блок питания, указано на блоках питания. Тип и количество блоков питания определяется заказным обозначением.

ВНИМАНИЕ! ПОДАЧА НАПРЯЖЕНИЯ 220 В (AC/DC) НА ВХОД ПИТАНИЯ 24 В (DC) или 24/48 В (DC) ПРИВЕДЕТ К НЕИСПРАВНОСТИ УСТРОЙСТВА.

Схемы подключения электропитания различных исполнений по питанию и соответствующая маркировка блоков питания приведена на рисунках ниже.

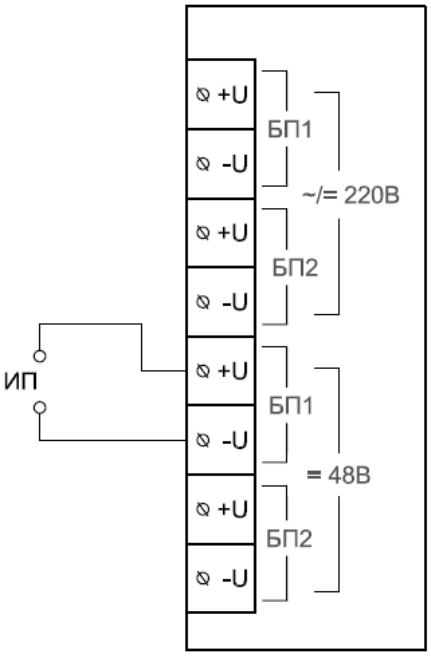


а) Схема подключения питания

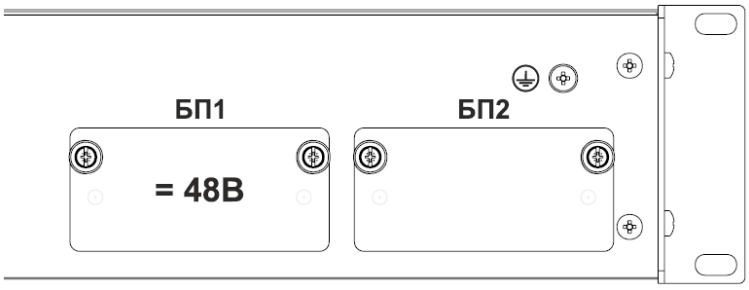


б) Маркировка блоков питания

Рисунок 28 – Схема подключения питания исполнения LV и соответствующая маркировка БП1

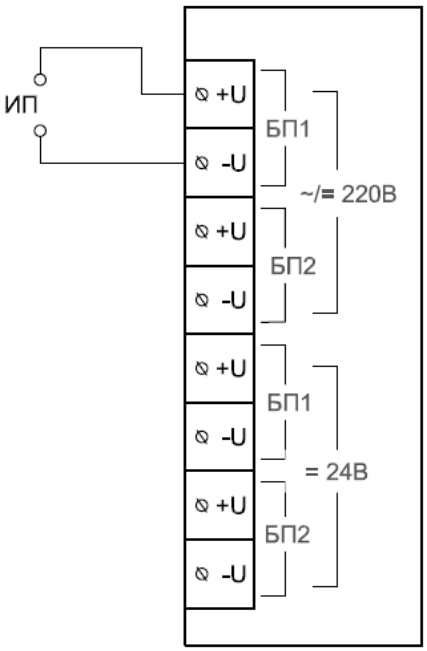


а) Схема подключения питания

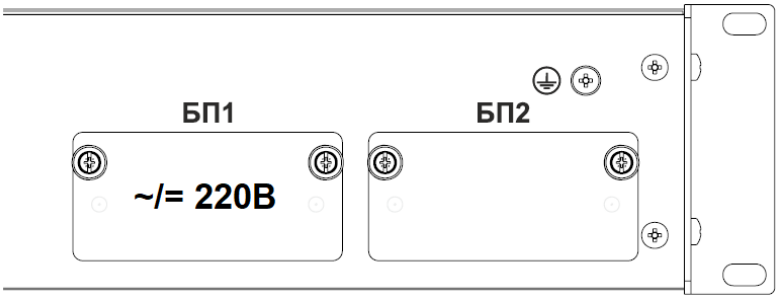


б) Маркировка блоков питания

Рисунок 29 – Схема подключения питания исполнения 24/48 и соответствующая маркировка БП1

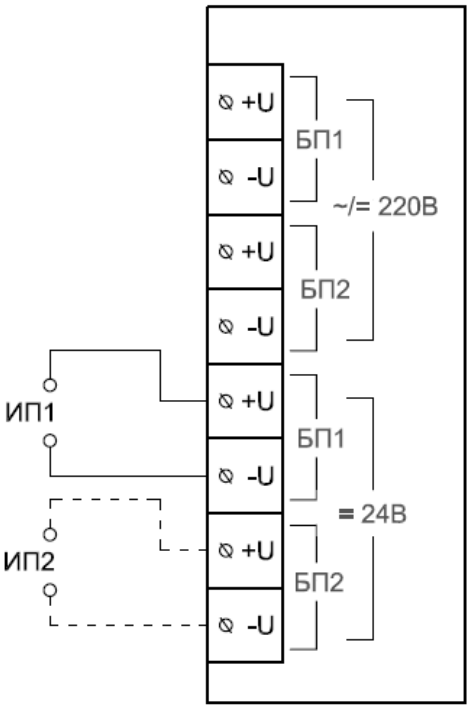


а) Схема подключения питания

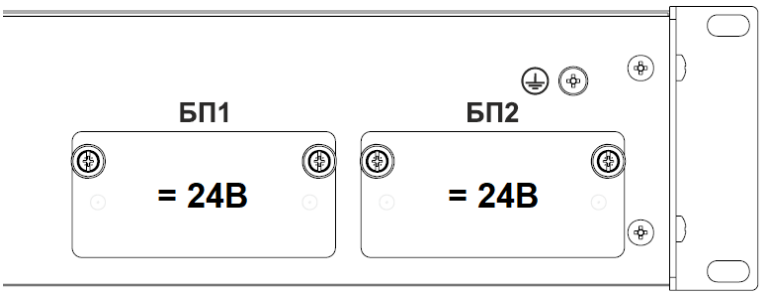


б) Маркировка блоков питания

Рисунок 30 – Схема подключения питания исполнения HV и соответствующая маркировка БП1

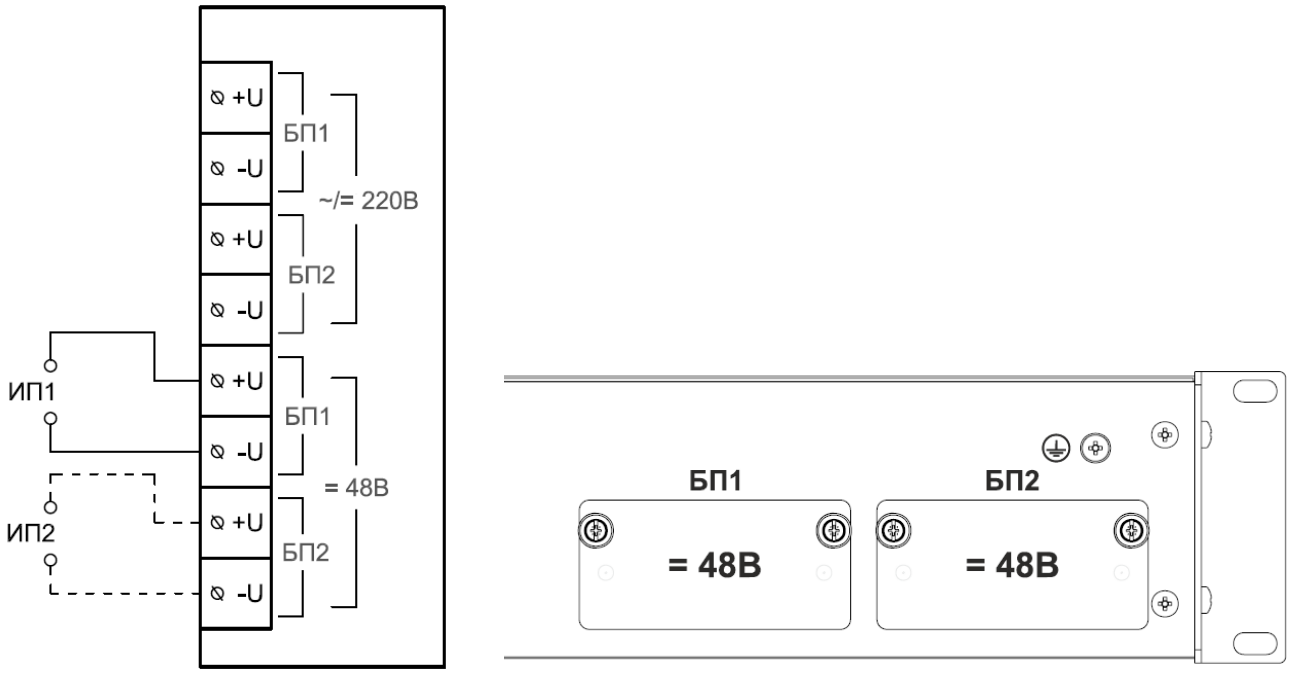


а) Схема подключения питания



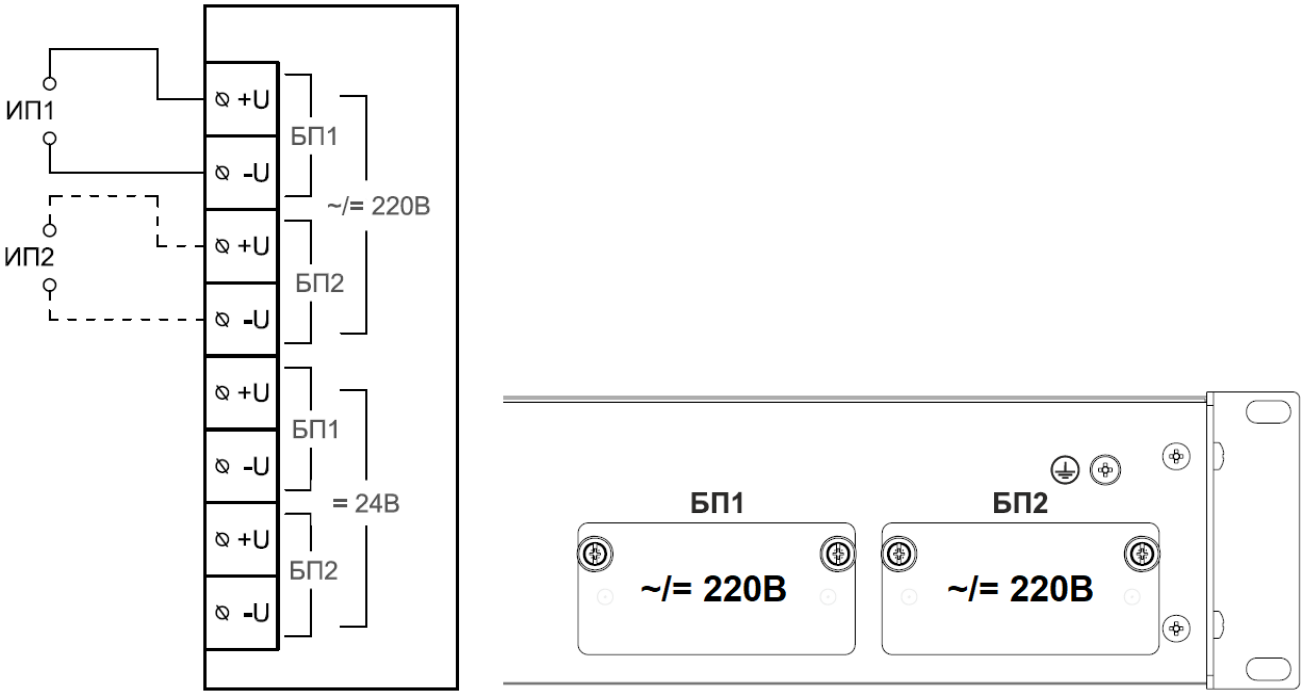
б) Маркировка блоков питания

Рисунок 31 – Схема подключения питания исполнения 2LV и соответствующая маркировка БП1 и БП2



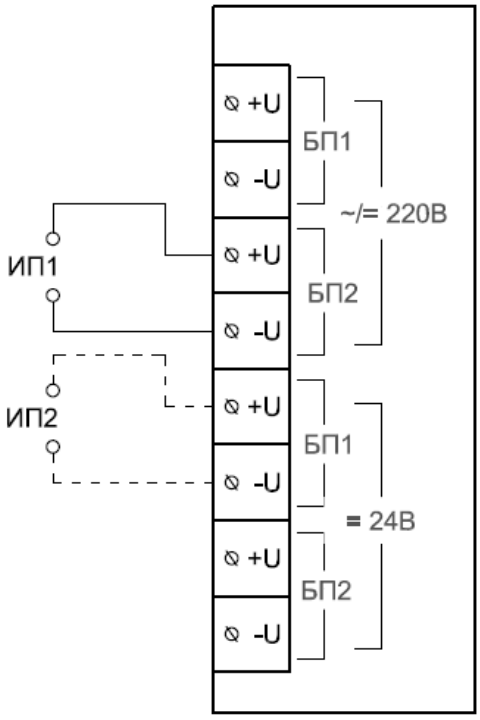
а) Схема подключения питания б) Маркировка блоков питания

Рисунок 32 – Схема подключения питания исполнения 24/48-24/48 и соответствующая маркировка БП1 и БП2

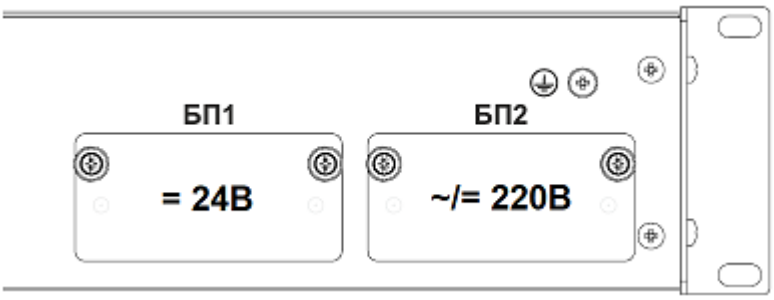


а) Схема подключения питания б) Маркировка блоков питания

Рисунок 33 – Схема подключения питания исполнения 2HV и соответствующая маркировка БП1 и БП2

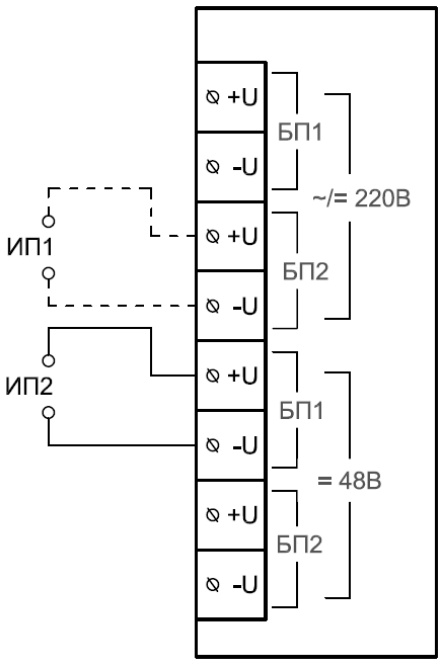


а) Схема подключения питания

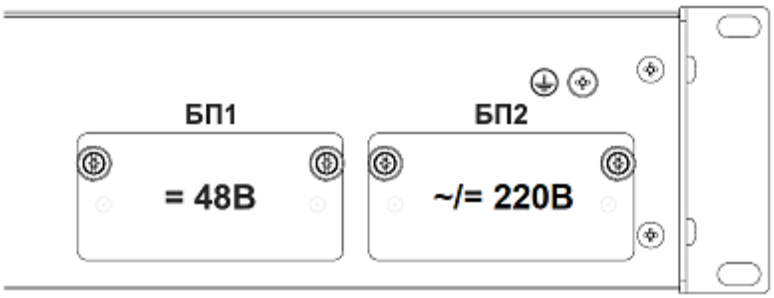


б) Маркировка блоков питания

Рисунок 34 – Схема подключения питания исполнения LV-HV и соответствующая маркировка БП1 и БП2



а) Схема подключения питания



б) Маркировка блоков питания

Рисунок 35 – Схема подключения питания исполнения 24/48-HV и соответствующая маркировка БП1 и БП2

7.2.6.3 Подключение цепей сигнализации

Внешний вид клемм для подключения цепей сигнализации представлен на рисунке ниже.

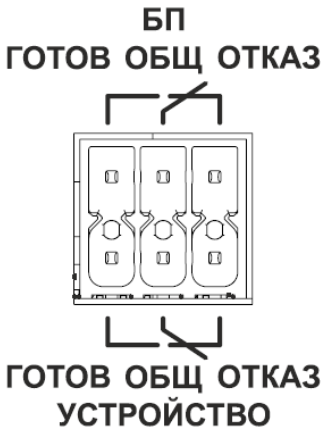


Рисунок 36

Подключение цепей сигнализации представлено на схемах ниже.

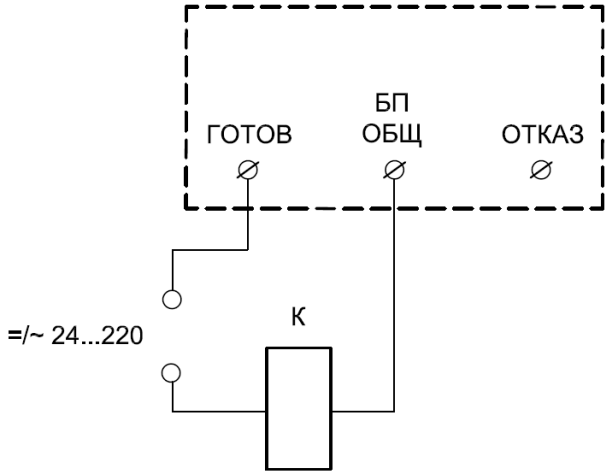


Рисунок 37 – Схема подключения цепей для проверки состояния каналов питания

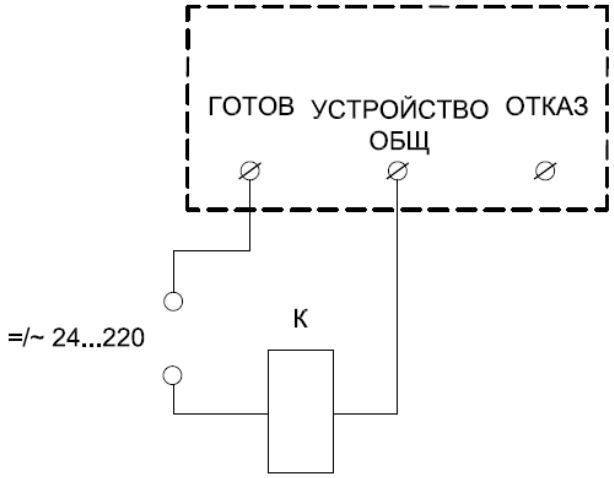


Рисунок 38 – Схема подключения цепей для диагностики исправности маршрутизатора

7.2.7 **Подключение к сети Ethernet**

Подключение к сети Ethernet осуществляется, используя промышленные коммутаторы, объединенные в локальную технологическую сеть с кольцевой или иной топологией (рекомендуется применять экранированные кабели и патч-корды).

7.2.7.1 Подключение оптоволоконных портов Ethernet

При подключении устройства по оптическому интерфейсу Ethernet используется две оптоволоконные линии. Одна из оптических линий используется для передачи от устройства 1 к устройству 2, а другая от устройства 2 к устройству 1, формируя, таким образом, полнодуплексную передачу данных.

Необходимо соединить Tx-порт (передатчик) устройства 1 с Rx-портом (приемник) устройства 2, а Rx-порт устройства 1 с Tx-портом устройства 2. При подключении кабеля

рекомендуется обозначить две стороны одной и той же линии одинаковой буквой (A-A, B-B, как показано ниже).

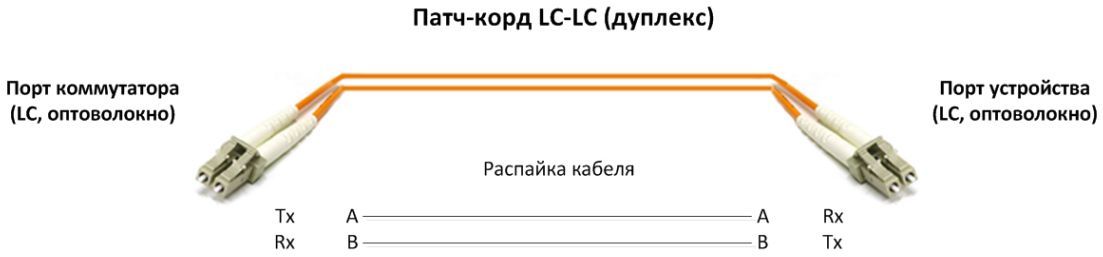


Рисунок 39 – Схема подключения оптоволоконного кабеля

ВНИМАНИЕ! УСТРОЙСТВО ЯВЛЯЕТСЯ ПРОДУКТОМ КЛАССА CLASS 1 LASER/LED. ИЗБЕГАЙТЕ ПРЯМОГО ПОПАДАНИЯ В ГЛАЗ ИЗЛУЧЕНИЯ LASER/LED.

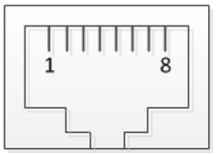
7.2.7.2 Подключение Ethernet-портов 10/100 BaseT(X)

Порты 10/100BaseTX, расположенные на передней панели, используются для подключения Ethernet-устройств.

На рисунке ниже схема расположения контактов для портов MDI (подключение устройств пользователя) и MDI-X (подключение коммутаторов/концентраторов), а также показана распайка прямого и перекрестного Ethernet-кабелей.

Таблица 27 – Назначение контактов

Контакт	Сигнал
порт MDI	
1	Tx+
2	Tx-
3	Rx+
6	Rx-
порт MDI-X	
1	Rx+
2	Rx-
3	Tx+
6	Tx-



8-контактный порт RJ45

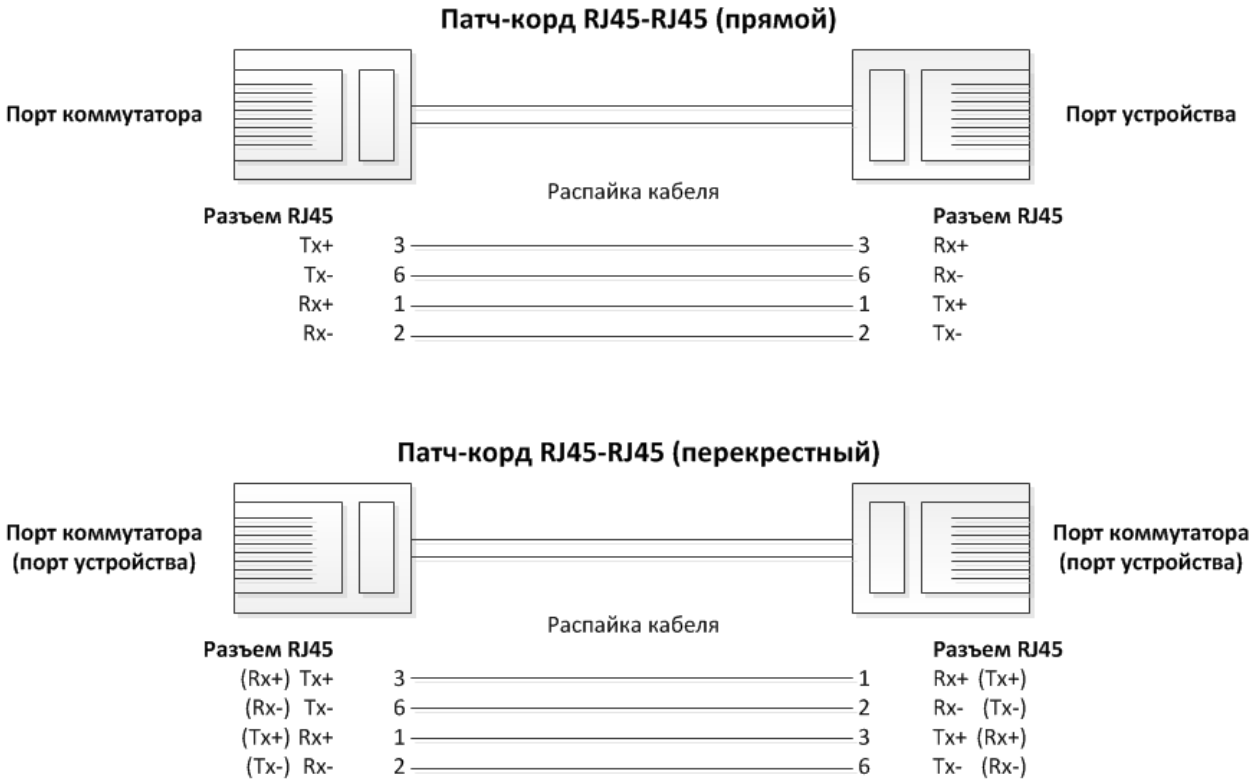


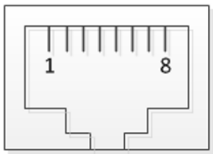
Рисунок 40 – Схема соответствия контактов

7.2.7.3 Подключение Ethernet-порта 1000BaseT(X)

Данные с порта 1000BaseT(X) передаются по дифференциальной сигнальной паре TRD+/- с помощью медных проводов.

Таблица 28 – Назначение контактов

Контакт	Сигнал
порт MDI/MDI-X	
1	TRD (0) +
2	TRD (0) -
3	TRD (1) +
4	TRD (2) +
5	TRD (2) -
6	TRD (1) -
7	TRD (3) +
8	TRD (3) -



8-контактный порт RJ45

7.2.8 Горячая замена блока питания в модификации TOPAZ FW MX710

При наличии двух встроенных блоков питания (далее – БП) устройство поддерживает функцию горячей замены БП. Для замены БП не требуется отсоединять цепи от клемм питания.

Примечание Для БП, рассчитанного на 220 В AC/DC, необходимо предварительно отключить питание. Включение/отключение питания производится путем перевода соответствующего автоматического выключателя БП в положение «включено»/«отключено».

Горячую замену БП необходимо осуществлять в следующем порядке:

1) в случае, если БП рассчитан на 220 В AC/DC, отключить питание заменяемого БП и убедиться в отсутствии напряжения на заменяемом БП (соответствующий индикатор **БП1** или **БП2** на передней панели устройства не горит);
2) открутить две фиксирующие гайки заменяемого БП;
3) извлечь заменяемый БП;
4) установить новый БП на место заменяемого;
5) убедиться, что новый БП вставлен до упора (дополнительные усилия прилагать нельзя);
6) вручную закрутить фиксирующие гайки нового БП;
7) в случае, если БП рассчитан на 220 В AC/DC, включить питание нового БП;
8) убедиться в наличии напряжения на новом БП (соответствующий индикатор **БП1** или **БП2** на передней панели устройства светится).

ПРИЛОЖЕНИЕ А

(Назначение контактов и портов)

Таблица А.1 – Обозначения клемм и портов TOPAZ FW MX240, TOPAZ FW MX681

Обозначение	Описание
Разъем T-BUS	
-24	Канал питания №1
+24	
Клеммный блок	
-24	Канал питания №2
+24	
Питание напряжением переменного тока	
~ 220 В	Клеммы питания 220 В
Заземление	
⏚	клемма заземления
Интерфейс конфигурирования	
USB	USB порт для подключения через консоль
Интерфейс Ethernet	
LANn	Порт Ethernet, где n номер порта

Таблица А.2 – Назначение клемм и портов TOPAZ FW MX710

Обозначение			Назначение
Каналы питания 24 В			
~/= 220V	**БП1**	**+U**	Вход от источника питания 220В №1
		-U	
	БП2	**+U**	Вход от источника питания 220В №2
		-U	
Каналы питания 220 В			
= 24V	**БП1**	**+U**	Вход от источника питания 24В №1
		-U	
	БП2	**+U**	Вход от источника питания 24В №2
		-U	
Реле сигнализации питания			
БП ОБЩ			Общий контакт
ГОТОВ			Нормально разомкнутый контакт (отсутствие неисправностей)
ОТКАЗ			Нормально замкнутый контакт (наличие неисправностей)
Реле сигнализации работы устройства			
УСТРОЙСТВО ОБЩ			Общий контакт
ГОТОВ			Нормально разомкнутый контакт (отсутствие неисправностей)
ОТКАЗ			Нормально замкнутый контакт реле (наличие неисправностей)
Порты			
КОНСОЛЬ			Порт конфигурирования
			Порт Ethernet

ПРИЛОЖЕНИЕ Б

(Назначение кнопок и индикаторов)

Таблица Б.1 – Обозначения кнопок и индикаторов

Обозначение	Описание
Кнопки (в наличии RS и RB)	
RS	Перезагрузка устройства
RB	Активация загрузчика с SD карты, при одновременном нажатии с кнопкой RS
Кнопки (в наличии RB)	
RB	Активация загрузчика с SD-карты
Индикаторы	
PWR	Наличие питания
RDY	Состояние готовности устройства

Таблица Б.2 – Назначение светодиодных индикаторов TOPAZ FW MX710

Наименование индикатора		Режим работы	Описание
Модули питания и ЦП			
ГОТОВ		не светится	Устройство не работает
		мигает	Устройство функционирует нормально
АВАРИЯ		светится непрерывно	Устройство неисправно
		не светится	Устройство исправно
КОНСОЛЬ		светится непрерывно	Наличие подключения к устройству
		не светится	Отсутствует подключение
СВЯЗЬ	**П1**	светится непрерывно	Наличие подключения к порту П1
		не светится	Отсутствует подключение к порту П1
	П2	светится непрерывно	Наличие подключения к порту П2
		не светится	Отсутствует подключение к порту П2
	П3	светится непрерывно	Наличие подключения к порту П3
		не светится	Отсутствует подключение к порту П3
	П4	светится непрерывно	Наличие подключения к порту П4
		не светится	Отсутствует подключение к порту П4
SFP	**П1**	светится непрерывно	Наличие подключения к порту П1 (SFP)
		не светится	Отсутствует подключение к порту П1 (SFP)
	П2	светится непрерывно	Наличие подключения к порту П2 (SFP)
		не светится	Отсутствует подключение к порту П2 (SFP)
	П3	светится непрерывно	Наличие подключения к порту П3 (SFP)
		не светится	Отсутствует подключение к порту П3 (SFP)
	П4	светится непрерывно	Наличие подключения к порту П4 (SFP)
		не светится	Отсутствует подключение к порту П4 (SFP)
БП1		светится непрерывно	Наличие питания от блока питания 1
		не светится	Отсутствие питания от блока питания
БП2		светится непрерывно	Наличие питания от блока питания 2
		не светится	Отсутствие питания от блока питания

ПРИЛОЖЕНИЕ В

(Внешний вид устройства)

Рисунок В.1 – Внешний вид TOPAZ FW MX240 E6 2GTx-4Tx-2LV

Рисунок В.2 – Внешний вид TOPAZ FW MX710

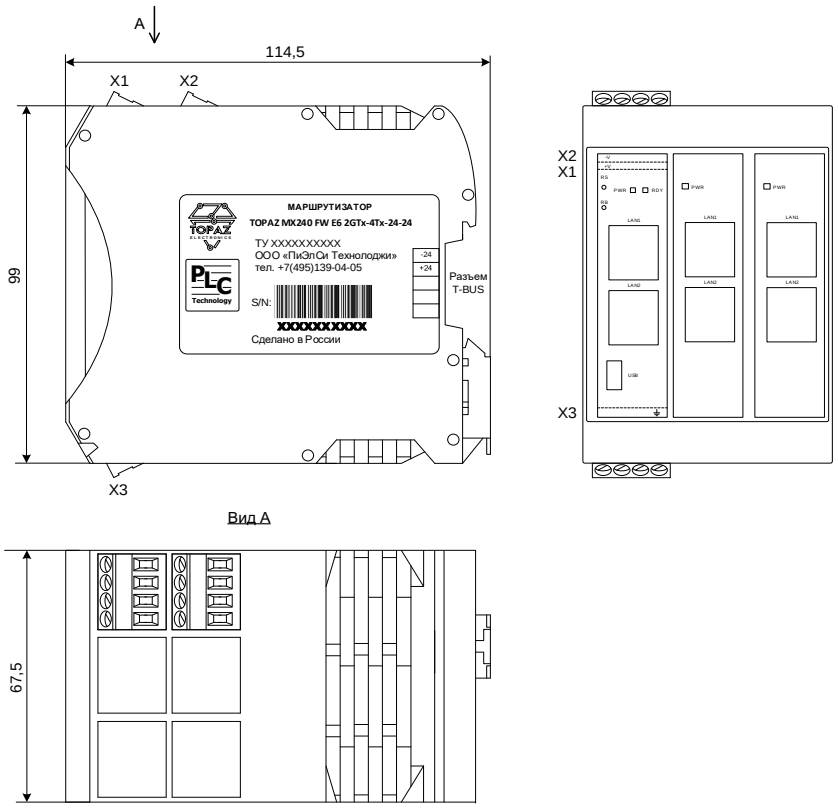


Рисунок В.3 – Габаритные размеры TOPAZ FW MX240 E6 (2GTx-4Tx)

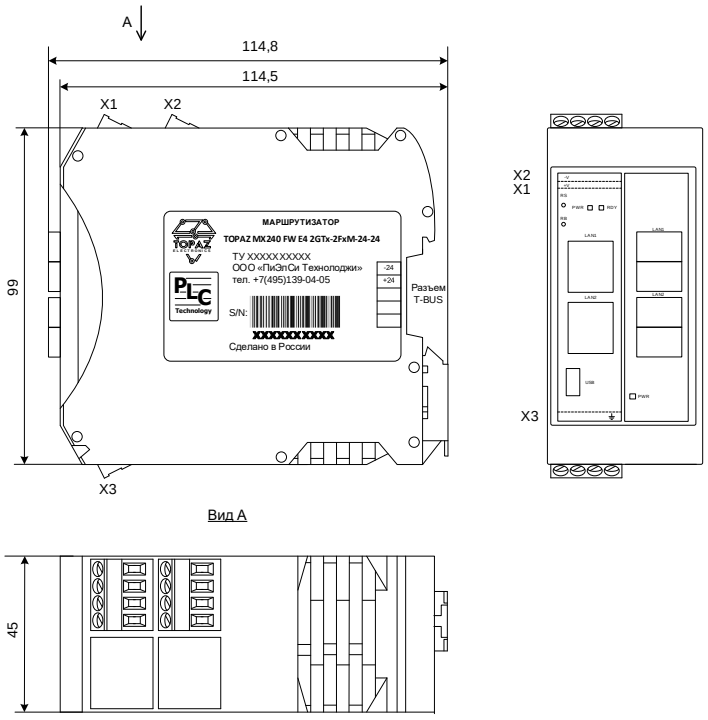


Рисунок В.4 – Габаритные размеры устройств
TOPAZ FW MX240 E4 2GTx-2FxM-2LV, TOPAZ FW MX240 E4 2GTx-2FxS-2LV

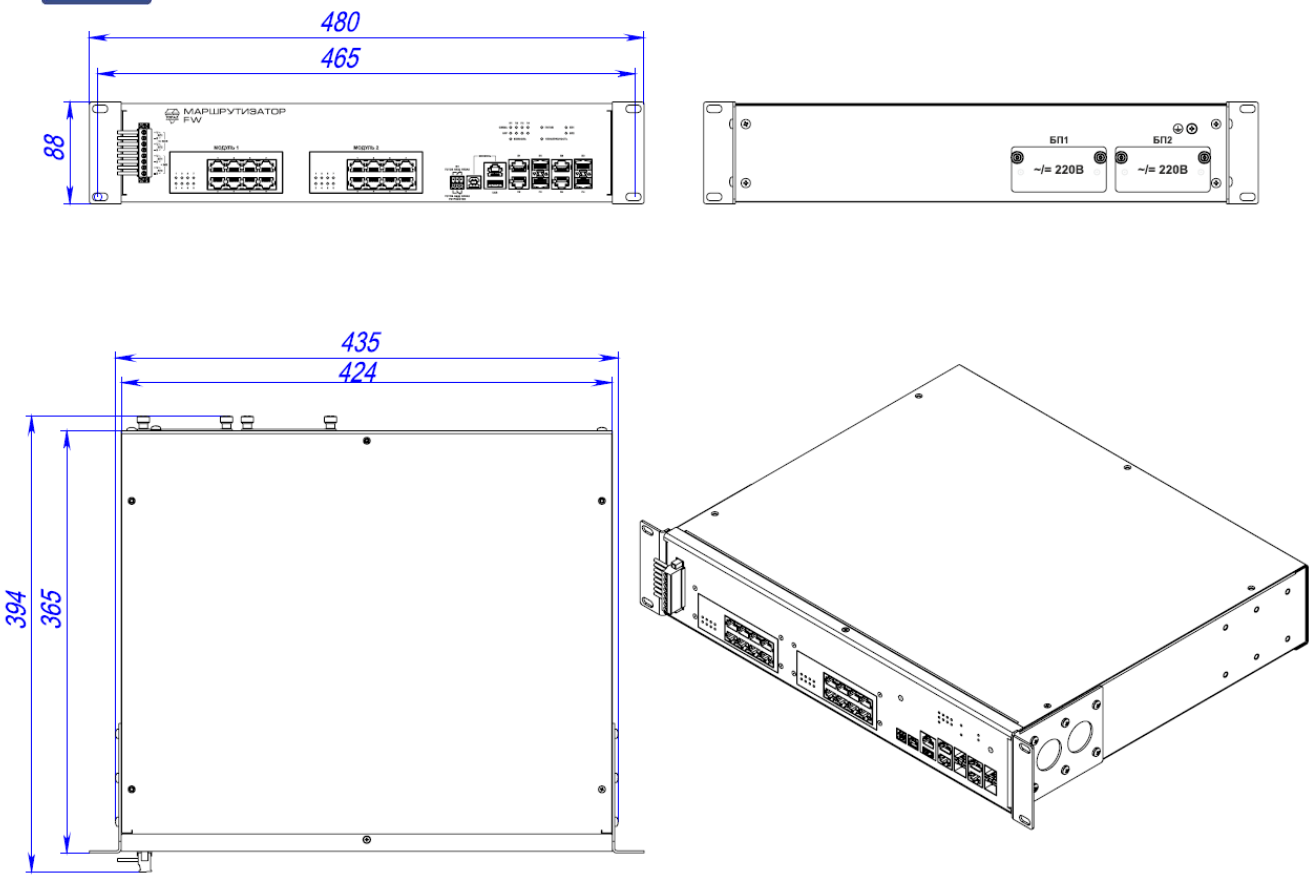


Рисунок В.5 – Габаритные размеры устройств TOPAZ FW MX710

ПРИЛОЖЕНИЕ Г

(Подключение к устройству с помощью утилиты PuTTY)

Утилита PuTTY – одна из распространенных бесплатных программ, не требующая установки. В данном разделе приведено описание подключения к устройству с помощью данной утилиты.

Сайт разработчика:
http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html.
Ссылка непосредственно исполняемый файл программы:
https://the.earth.li/~sgtatham/putty/latest/x86/putty.exe.

Подключение через серийный порт

После запуска программы PuTTY откроется окно настройки, где во вкладке **Session** необходимо выбрать тип соединения **Serial** и его основные параметры (номер виртуального порта будет отличаться от приведенного в примере в зависимости от вашей системы):

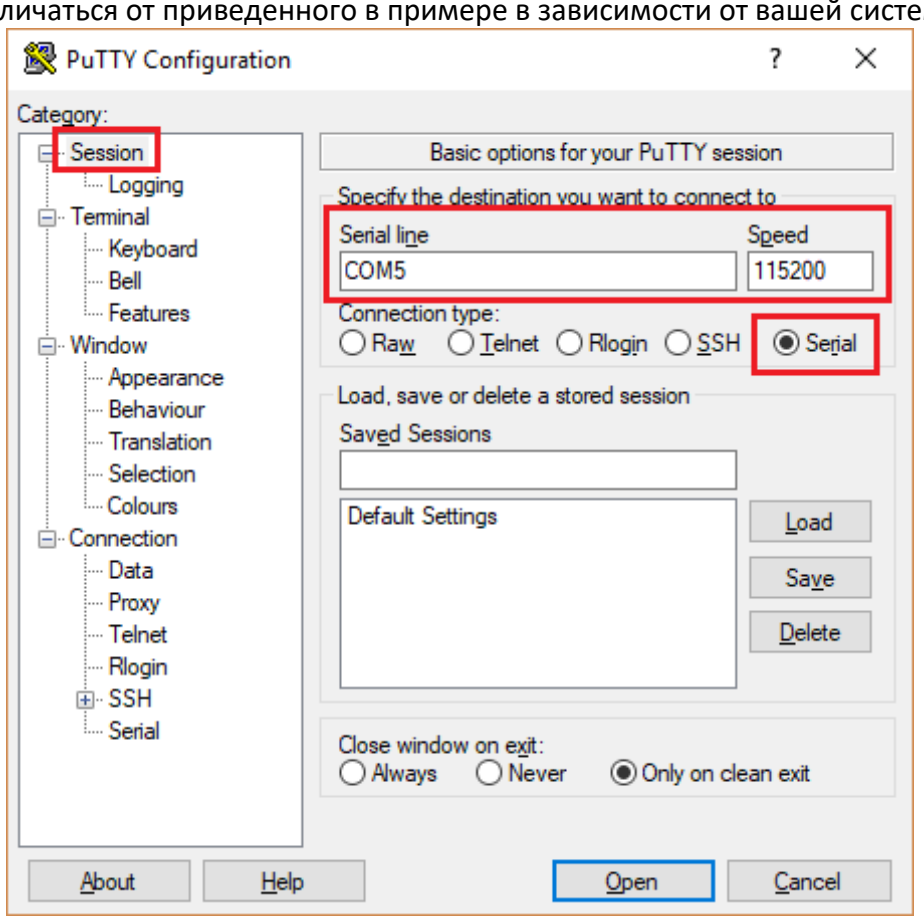


Рисунок Г.1 – Задаваемые настройки раздела Session (сессия)

В настройках соединения (**Connection**) – выбрать последовательный порт (**Serial**) и установить параметры соединения согласно таблице 11:

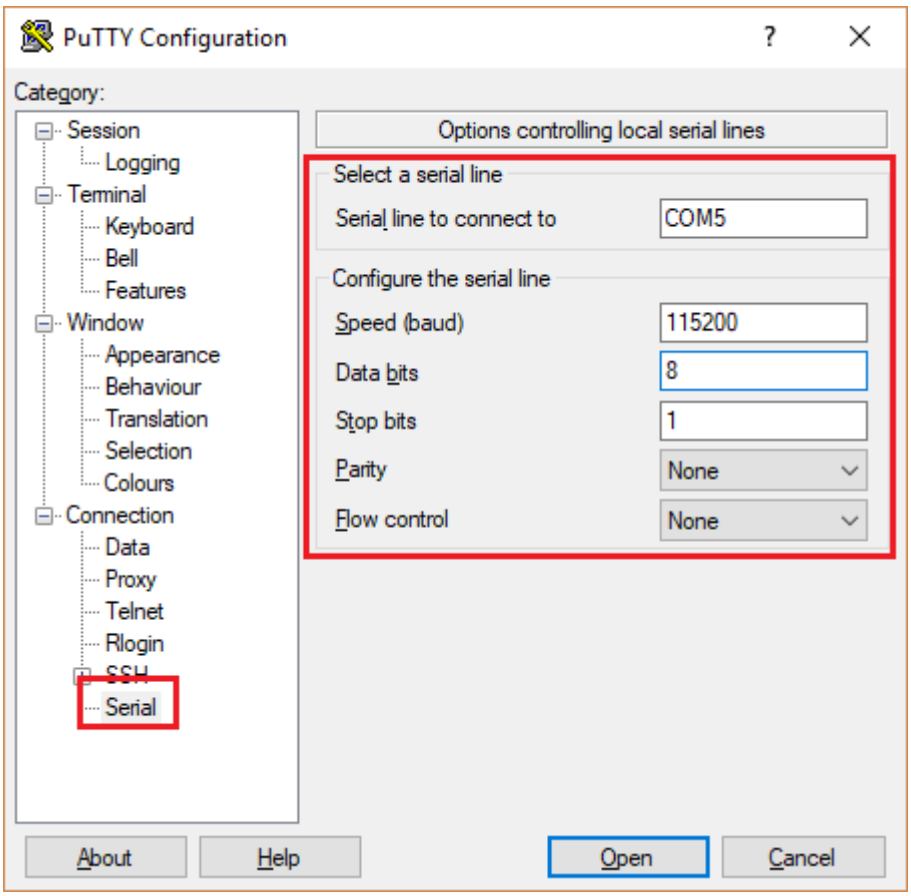


Рисунок Г.2 – Задаваемые настройки раздела Serial (серийный порт)

После настройки параметров последовательного порта, необходимо нажать кнопку «Открыть» (Open) для установки соединения и вызова окна консоли.

Подключение через Ethernet порт

Для подключения к устройству по протоколу SSH, во вкладке **Session** необходимо выбрать тип соединения **SSH** и его основные параметры:

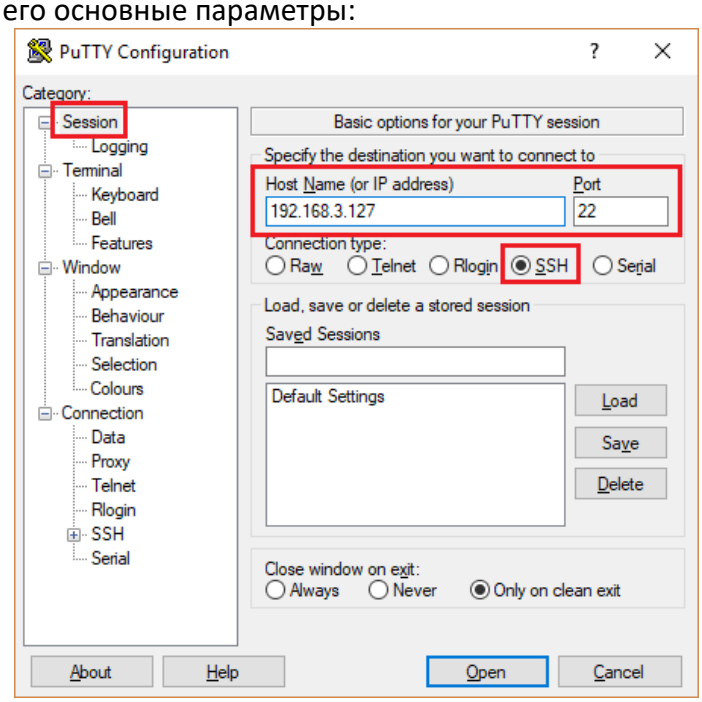


Рисунок Г.3 – Задаваемые настройки раздела Session (сессия)

После настройки параметров последовательного порта, необходимо нажать кнопку «Открыть» (Open) для установки соединения и вызова окна консоли.