Kaspersky SD-WAN

Руководство по настройке демонстрационного стенда Kaspersky SD-WAN в среде VMware

Часть 2

Настройка сценариев классификации, приоритезации и управления трафиком, построения Full-Mesh и Partial-Mesh топологий, обновление CPE устройств, резервирование CPE с помощью VRRP.

kaspersky

Содержание

kaspersky

1. **Kaspersky SD-WAN.**

Решение Kaspersky SD-WAN используется для построения программно-определяемых распределенных сетей (англ. Software Defined WAN или SD-WAN) для маршрутизации сетевого трафика по каналам сети передачи данных с применением технологии SDN (Software Defined Networking). В сетях SD-WAN наиболее эффективные пути маршрутизации трафика определяются автоматически.

Технология SDN подразумевает разделение уровня управления сетью (англ. Control Plane) и уровня передачи данных (англ. Data Plane). Уровень управления контролирует передачу пакетов по сети через телекоммуникационное оборудование, установленное на площадке клиента (англ. Customer Premises Equipment, или устройства CPE). Передача пакетов через устройства CPE осуществляется на уровне передачи данных.

В сетях, построенных с применением технологии SDN, уровень управления переносится в централизованный контроллер SD-WAN. Данный контроллер взаимодействует с устройствами CPE, составляющими уровень передачи данных, а также с SD-WAN оркестратором, который используется для управления сетью SD-WAN с помощью веб-интерфейса.

Решение Kaspersky SD-WAN предназначено для операторов связи, компаний, имеющих крупную филиальную сеть, и используется для замены стандартных маршрутизаторов в распределенных сетях.

Решение Kaspersky SD-WAN обладает следующими основными характеристиками:

- Работа на основе проводных и беспроводных сетей любого типа.
- Использование несколько виртуальных каналов для обеспечения высокой доступности сети и балансировки трафика.
- Коррекция ошибок при передаче данных.
- Интеллектуальное управление трафиком.
- Автоматическая настройка устройств CPE с использованием концепции Zero Trust Provisioning (ZTP).
- Централизованное управление и мониторинг.

1.1. **Архитектура решения Kaspersky SD-WAN.**

Краткое описание основных компонентов решения Kaspersky SD-WAN:

- SD-WAN оркестратор. Предоставляет единый графический веб-интерфейс управления, отвечает за управление сервисами SD-WAN сети и содержит инвентаризационную базу CPE устройств.
- SD-WAN контроллер. Управляет наложенной сетью (англ. Overlay Network), обеспечивает построение топологии сети и создание транспортных сервисов внутри наложенных туннелей. Поддерживает транспортные сервисы L2 Point-to-Point (P2P), Point-to-Multipoint (P2M), Multipoint-to-Multipoint (M2M) и L3 VPN. Управляет устройствами CPE и шлюзами SD-WAN по протоколу OpenFlow. Определяет распределение трафика между туннелями, выполняет мониторинг качества соединения и автоматическое переключение трафика на резервный туннель в случае возникновения проблем на основном. Контроллер находится под управлением SD-WAN оркестратора.
- SD-WAN шлюзы. Объединяют CPE устройства в единую сеть. Наложенные туннели терминируются на SD-WAN шлюзах, после чего трафик передается дальше в соответствии с топологией сети.
- CPE устройства или Kaspersky Edge Service Router (KESR). Телекоммуникационное оборудование, которое подключается к шлюзам SD-WAN с помощью наложенных туннелей и образует SDN-фабрику в виде наложенной сети.

Архитектура решения Kaspersky SD-WAN представлена на рисунке 1.

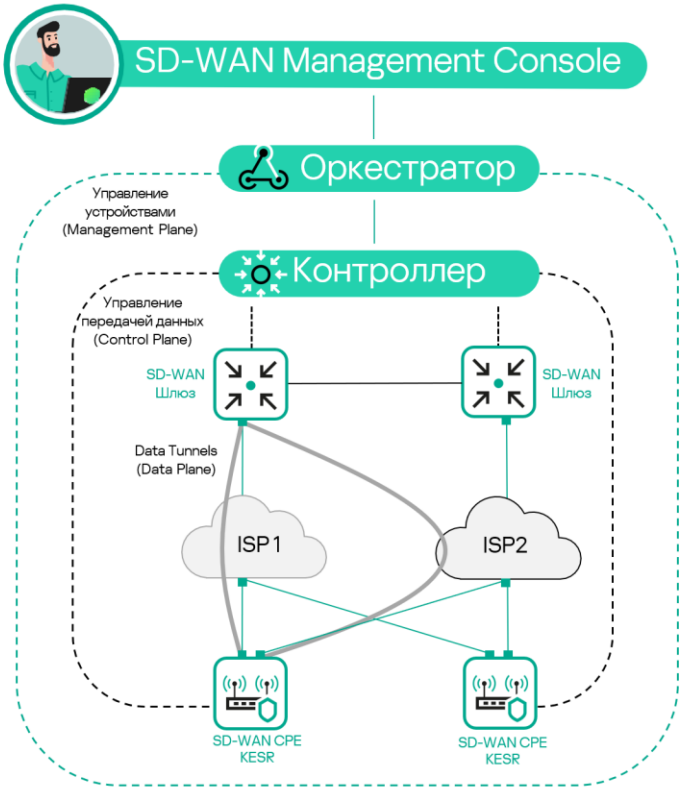


Рисунок 1. Архитектура решения Kaspersky SD-WAN.

kaspersky

2. **Описание схемы демонстрационного стенда Kaspersky SD-WAN.**

Все компоненты демонстрационного стенда Kaspersky SD-WAN развернуты в среде виртуализации VMware.

Развертывание и базовая настройка демонстрационного стенда описаны в первой части документа Proof of Concept Руководство по настройке демонстрационного стенда Kaspersky SD-WAN Часть 1.

На виртуальном хосте orc1 развернуты Docker контейнеры решения Kaspersky SD-WAN, включая оркестратор, контролер и систему мониторинга Zabbix.

Логическая схема демонстрационного стенда Kaspersky SD-WAN представлена на рисунке 2. Демонстрационный стенд включает в себя:

- Площадка DC с сетевыми сегментами dc-lan1 и oob, подключенными к маршрутизатору R13. Виртуальная машина SD-WAN оркестратора orc1 размещена в сегменте oob, сервер srv1 с WWW службой размещен в сегменте dc-lan1.
- На границе DC размещены два маршрутизатора R11 и R12, за которыми размещены два SD-WAN шлюза: vGW-11 и vGW-12. Внутренние (lan) интерфейсы R13, vGW-11 и vGW-12 подключены к сетевому сегменту dc-perim.
- Маршрутизаторы R11 и R12 выполняют функцию SNAT для vGW-11 и vGW-12 и DNAT для портов, указанных в Таблице №1.
- Маршрутизатор R14 выполняет SNAT, роль шлюза по умолчанию для R13, и выход в Интернет для хоста orc1. R14 выполняет DNAT для хоста orc1 для портов, указанных в Таблице №1 для Docker контейнеров SD-WAN оркестратора и SD-WAN контроллера.
- Хост ISP эмулирует подключение к сети Интернет / операторам связи ISP1 – ISP8.
- Для подключения CPE устройств SD-WAN шлюзы должны быть доступны по определённому набору портов, перечисленных в Таблице №1.
- Устройство vCPE-3 представляет собой пример подключения удаленной площадки с одним CPE устройством, подключенным к двум операторам связи.
- Устройство vCPE-4 представляет собой пример будущего, не рассматриваемой в рамках текущего стенда, подключения удаленной площадки с универсальным uCPE устройством.
- Шлюзы vCPE-51 и vCPE-52 представляют собой пример подключения удаленной площадки с двумя CPE устройствами. Для отказоустойчивости используется протокол VRRP.

2.1. **Схема демонстрационного стенда.**

Схема демонстрационного стенда Kaspersky SD-WAN представлена на рисунке 2.

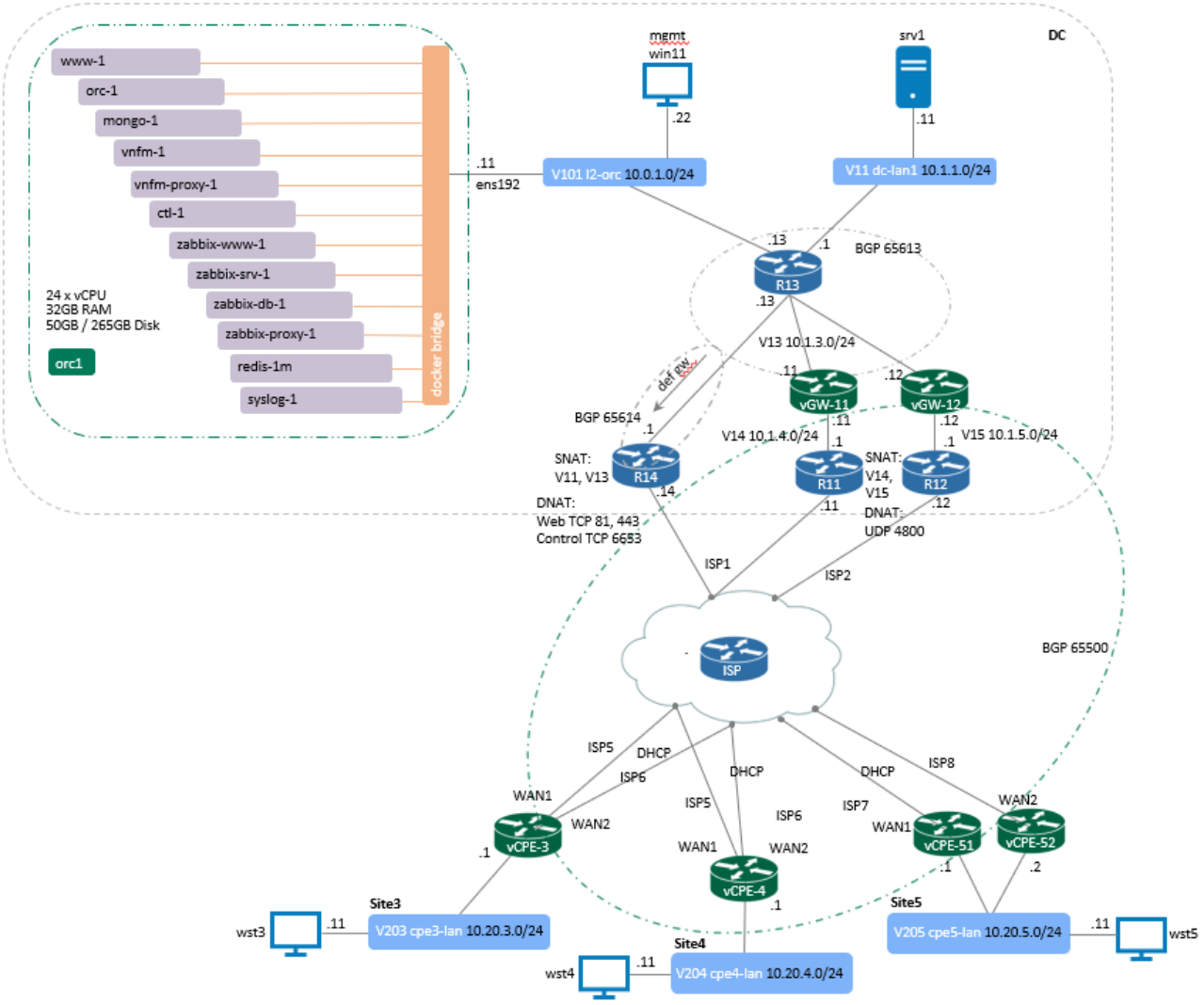


Рисунок 2 – Демонстрационный стенд Kaspersky SD-WAN 2.0

2.2. **Сетевые порты, используемые центральными компонентами решения.**

Компонент	**Порт**	**Назначение**
SD-WAN оркестратор	TCP 82 TCP 85 TCP 443	Доступ к веб- интерфейсу CPE через веб-интерфейс оркестратора. Доступ к веб- интерфейсу Zabbix. Доступ к веб- интерфейсу оркестратора.
SD-WAN контроллер	TCP 6653-6656	Подключение SD-WAN шлюзов и CPE устройств к контроллеру по TLS. CPE устройство подключается каждым wan интерфейсом к отдельному порту контроллера: • sdwan0 - 6653 • sdwan1 - 6654 • и т.д.
SD-WAN шлюзы	UDP 4800	Дата трафик.

Таблица 1. – Сетевые порты для взаимодействия SD-WAN шлюзов и CPE устройств с центральными компонентами решения, и доступ к веб- интерфейсу оркестратора для администрирования решения.

2.3. **План IP адресации.**

Данный IP план соответствует схеме из пункта 2.1 в случае использования других адресов требуется изменить план и все настройки SD-WAN в дальнейших шагах.

Имя	**Операционная система**	**IP адрес**	**Назначение**	**Минимальные ресурсы**
orc1	Ubuntu 20.04.06 LTS Server	10.0.1.11	На хосте развернуты Docker контейнеры: www-1, orc-1, redis-1m, mongo-1, vnfm-1, vnfm-proxy-1, ctl, zabbix-www-1, zabbix-srv-1, zabbix-prx-1, zabbix-db-1, syslog-1	24 x vCPU, 32 GB RAM
vGW-11	CPEOS	wan 10.1.4.11 lan 10.1.3.11	SD-WAN шлюз	4 x vCPU, 2 GB RAM
vGW-12	CPEOS	wan 10.1.5.12 lan 10.1.3.12	SD-WAN шлюз	4 x vCPU, 2 GB RAM
vCPE-3	CPEOS	wan DHCP lan 10.20.3.1	CPE	4 x vCPU, 2 GB RAM
vCPE-4	CPEOS	wan DHCP lan 10.20.4.1	CPE	4 x vCPU, 2 GB RAM
vCPE-51	CPEOS	wan DHCP lan 10.20.5.1	CPE	4 x vCPU, 2 GB RAM
vCPE-52	CPEOS	wan DHCP lan 10.20.5.2	CPE	4 x vCPU, 2 GB RAM
R11	CentOS 7	wan 10.50.1.11 lan 10.1.4.1	Пограничный маршрутизатор DC	2 x vCPU, 2 GB RAM
R12	CentOS 7	wan 10.50.2.12 lan 10.1.5.1	Пограничный маршрутизатор DC	2 x vCPU, 2 GB RAM
R13	CentOS 7	dc-perim 10.1.3.13 oob 10.0.1.13 dc-lan1 10.1.1.1	Маршрутизатор ядра DC	2 x vCPU, 2 GB RAM

Имя	Операционная система	IP адрес	Назначение	Минимальные ресурсы
R14	CentOS 7	wan 10.50.1.14 lan 10.1.3.1	Пограничный маршрутизатор DC, NAT	2 x vCPU, 2 GB RAM
ISP	CentOS 7	isp1 10.50.1.1 isp2 10.50.2.1 isp5 10.50.5.1 isp6 10.50.6.1 isp7 10.50.7.1 isp8 10.50.8.1	Эмуляция ISP1 – ISP8	2 x vCPU, 2 GB RAM
srv1	CentOS 7	10.1.1.11	Сервер WWW/DC	2 x vCPU, 4 GB RAM
wst3	CentOS 7	10.20.3.11	Рабочая станция Site3	2 x vCPU, 4 GB RAM
wst4	CentOS 7	10.20.4.11	Рабочая станция Site4	2 x vCPU, 4 GB RAM
wst5	CentOS 7	10.20.5.11	Рабочая станция Site5	2 x vCPU, 4 GB RAM
mgmt	Windows 11	10.0.1.10 10.1.1.10 10.1.3.10 10.50.1.10 10.20.3.10 10.20.4.10 10.20.5.10	Рабочая станция для управления демо стендом.	6 x vCPU, 6 GB RAM

2.4. **Версии программного обеспечения.**

Таблица №3 – Версии программного обеспечения Kaspersky SD-WAN, используемого в данном демонстрационном стенде:

Компонент SD-WAN	**Версия**
www	knaas-www:2.23.07.release.81.amd64_en-US_ru-RU
orc	knaas-orc:2.23.07.release.88.amd64_en-US_ru-RU
mongo	mongo:5.0.7.amd64
ctl	knaas-ctl:2.23.07.release.39.amd64_en-US_ru-RU
vnfm	knaas-vnfm:2.23.07.release.8.amd64_en-US_ru-RU
vnfm-proxy	knaas-vnfm-proxy:2.23.07.release.2.amd64_en-US_ru-RU
redis	redis:6.2.7.amd64
zabbix-www	zabbix-web-nginx-mysql:5.0.32.amd64
zabbix-proxy	zabbix-proxyr-mysql:5.0.32.amd64
zabbix-srv	zabbix-server-mysql:5.0.32.amd64
zabbix-db	mariadb:10.4.28.amd64
syslog	syslog-ng:3.30.1.amd64
vCPE	knaas-cpe_2.23.07.release.23.combined.amd64-legacy.qcow2
Хост orc1	Ubuntu 20.04.06 LTS Server
installer	knaas-installer_2.23.07.release.29.amd64_en-US_ru-RU.tar.gz

2.5. **Требования к аппаратным ресурсам решения Kaspersky SD-WAN.**

Таблица №4 - Требования к аппаратным ресурсам для управления до 50 CPE устройств.

Хост	**CPU (hyper-threading), cores**	**RAM, GB**	**Disk, GB, SSD** Используется в данной конфигурации / Рекомендуется
orc1	24	32	50 / 265

3. **Управление трафиком.**

Соединение между устройствами CPE устанавливается через туннели GENEVE, которые строятся поверх каналов передачи данных. Туннели (линки) являются однонаправленными, поэтому при соединении двух устройств CPE требуется входящий и исходящий туннель.

Совокупность туннелей, соединяющих два устройства CPE, является сегментом. Трафик может быть распределен по нескольким туннелям на устройстве CPE- отправителе в начале сегмента и передан устройству CPE- получателю в конце сегмента.

Маршруты, по которым трафик может быть передан в рамках одного сегмента, являются транспортными путями. Поддерживается использование следующих типов транспортных путей:

- Auto-SPF (Shortest-Path Forwarding). Автоматически рассчитываемый контроллером SD-WAN транспортный путь. Транспортные пути этого типа невозможно добавлять и удалять, а также изменять их параметры.
- Manual-TE (Traffic Engineering). Транспортный путь, который добавляется вручную. Для добавления транспортного пути этого типа требуется указать параметры туннелей, через которые транспортный путь будет проходить от устройства CPE в начале сегмента до устройства CPE в конце сегмента.
- Auto-TE. Автоматически рассчитываемый контроллером SD-WAN транспортный путь, учитывающий преднастроенные ограничения (англ. constraints). Ограничениями могут быть значения показателей мониторинга на туннелях, например, показатель уровня загрузки туннеля.

Транспортные пути имеют следующие параметры:

- Стоимость (англ. Path.cost). По умолчанию, является суммой стоимости всех туннелей, которые входят в транспортный путь. Поддерживается возможность ручного определения стоимости транспортных путей.
- Вес (англ. Path.weight).
- Административное состояние (англ. Path.admin.state). Задается вручную. Если этот параметр имеет значение down, транспортный путь не используется.
- Фактическое состояние (англ. Path.oper.state). Зависит от наличия или отсутствия возможности передачи трафика. Если этот параметр имеет значение down, транспортный путь не используется.

Один сегмент может содержать от 2 до 16 транспортных путей, при передаче трафика по умолчанию будет выбран наилучший транспортный путь с наименьшим значением атрибута стоимости. Если наилучший транспортный путь недоступен для передачи трафика по техническим причинам, выбирается другой транспортный путь с приближенным значением атрибута стоимости.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Туннели, сегменты и транспортные пути: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/250984.htm

3.1. **Балансировка нагрузки в режиме Active / Active.**

Kaspersky SD-WAN обеспечивает защиту от перерывов связи с устройствами CPE с помощью одновременного использования всех доступных каналов передачи данных. Поддерживаются следующие режимы резервирования каналов передачи данных: Active/Active и Active/Standby.

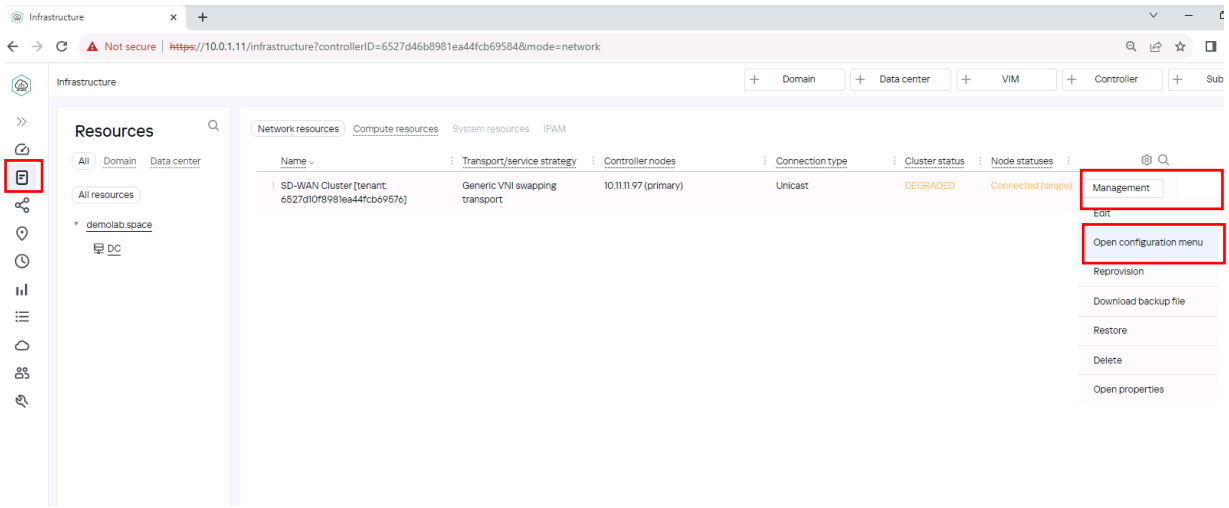
Для получения дополнительной информации о резервировании каналов связи обратитесь к Kaspersky SD-WAN Online Help > Резервирование каналов передачи данных между устройствами CPE: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/239053.htm

В данном сценарии рассматривается сценарий балансировки нагрузки между интерфейсами устройства vCPE-3. На устройстве vCPE-3 используется пара WAN интерфейсов в режиме Active / Active. Для балансировки нагрузки используется параметр Cost туннелей.

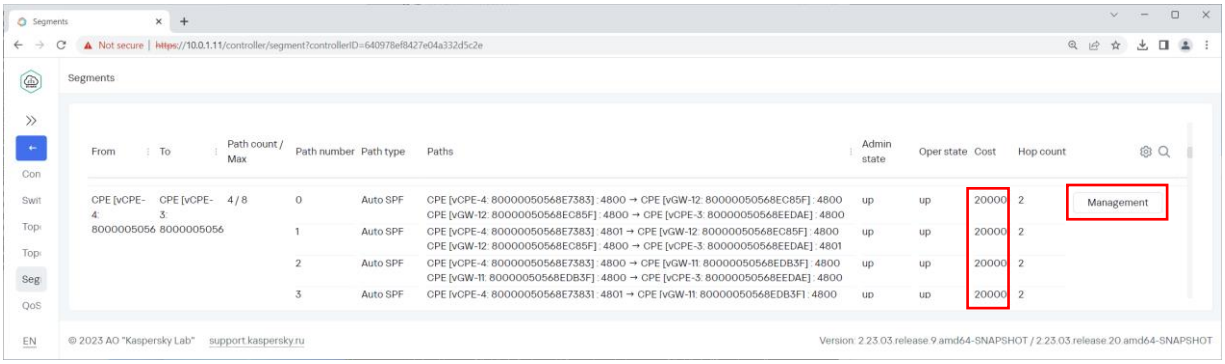
Для демонстрации балансировки трафика между vCPE-3 и vCPE-4 на рабочих станциях wst3 и wst4 используется генератор трафика iperf. Для проверки работы балансировки будет использована встроенная система мониторинга.

3.1.1. Просмотр построенных сегментов SD-WAN фабрики.

Для отображения перечня всех сегментов SD-WAN фабрики перейти в меню Infrastructure > SD-WAN Cluster > Management > Open configuration menu > Segments.

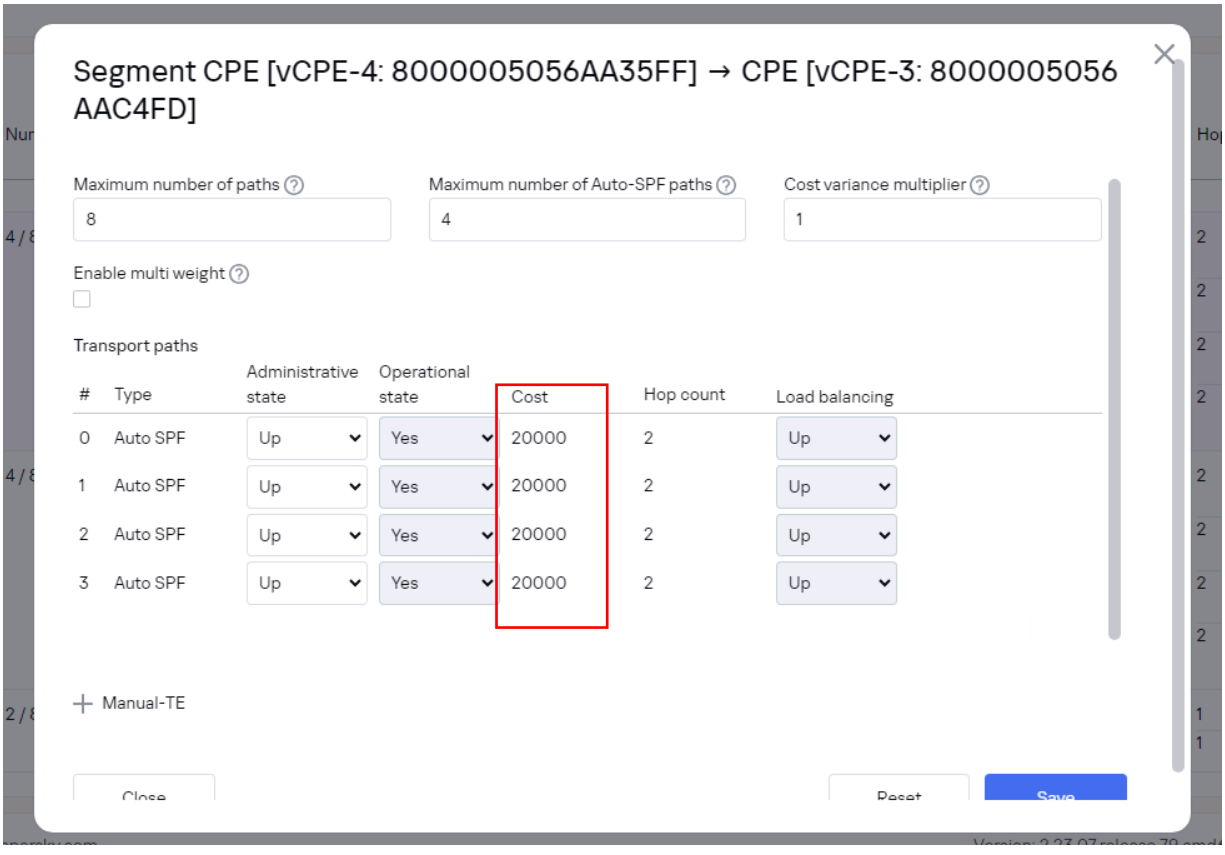


На скриншоте представлен пример сегмента между vCPE-4 и vCPE-3, состоящий из четырех транспортных путей (path) типа Auto SPF. Транспортные пути проходят через CPE с ролью Gateway: vGW-11 и vGW-12.



Стоимость (Cost) каждого из четырех путей равна, значение 20000, поэтому трафик равномерно балансируется между всеми транспортными путями в рамках сегмента между vCPE-4 и vCPE-3.
Балансировка осуществляется средствами протокола OpenFlow (группы типа Select).

Для получения дополнительной информации о параметрах балансировки нажать кнопку Management > Edit.



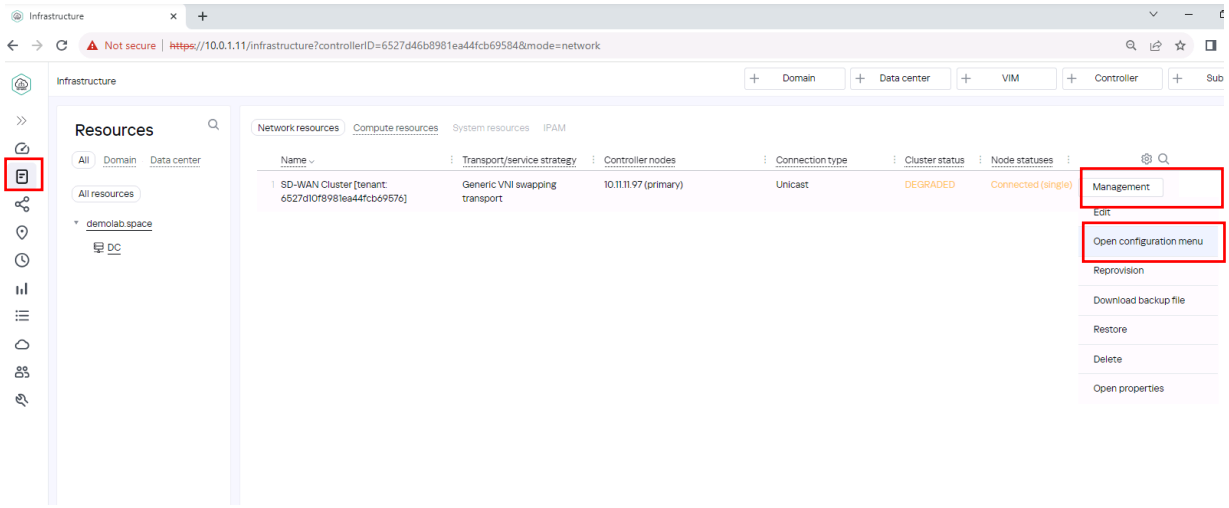
Контроллер заранее просчитывает все возможные транспортные пути, в том числе и резервные, например, если, фактическое количество транспортных путей больше, чем задано в параметре Maximum number of Auto-SPF paths для конкретного сегмента. Как только будет обнаружено событие отказа туннеля (линка) между CPE устройствами, туннель будет удален из топологии, а трафик перенаправлен на резервный транспортный путь.

3.1.2. Выбор режима балансировки.

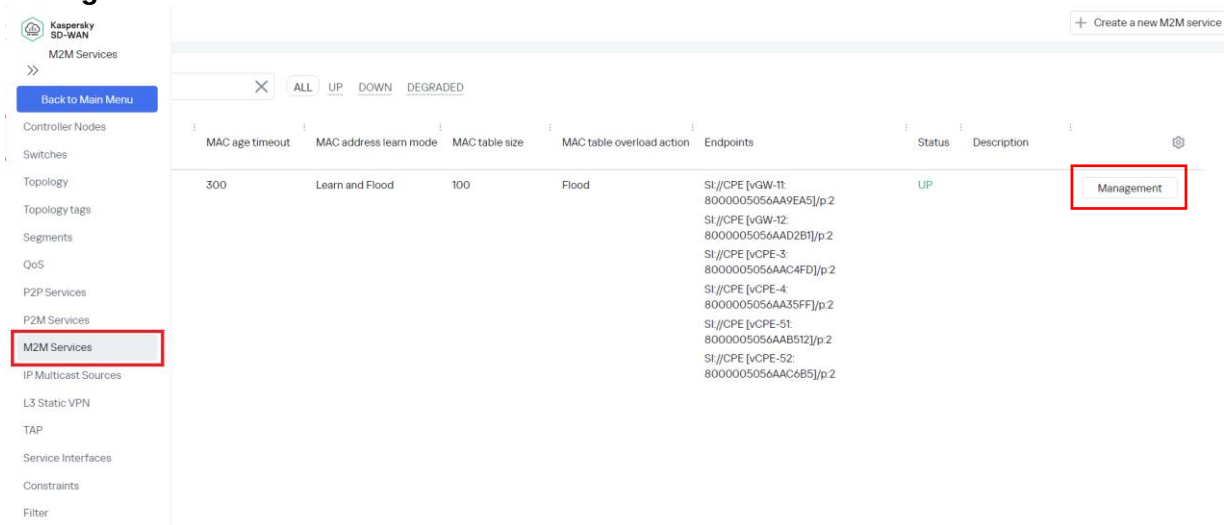
Доступные режимы балансировки:

- Per-flow. Балансировка по потокам (сессиям). При передаче потоки распределяются равномерно по туннелям.
- Per-packet. Балансировка по пакетам. При передаче пакеты распределяются равномерно по туннелям.
- Broadcast. Пакеты передаются одновременно во все туннели для исключения потерь.

To configure the balancing mode, go to **Infrastructure > SD-WAN Controller > Management > Open configuration menu**.



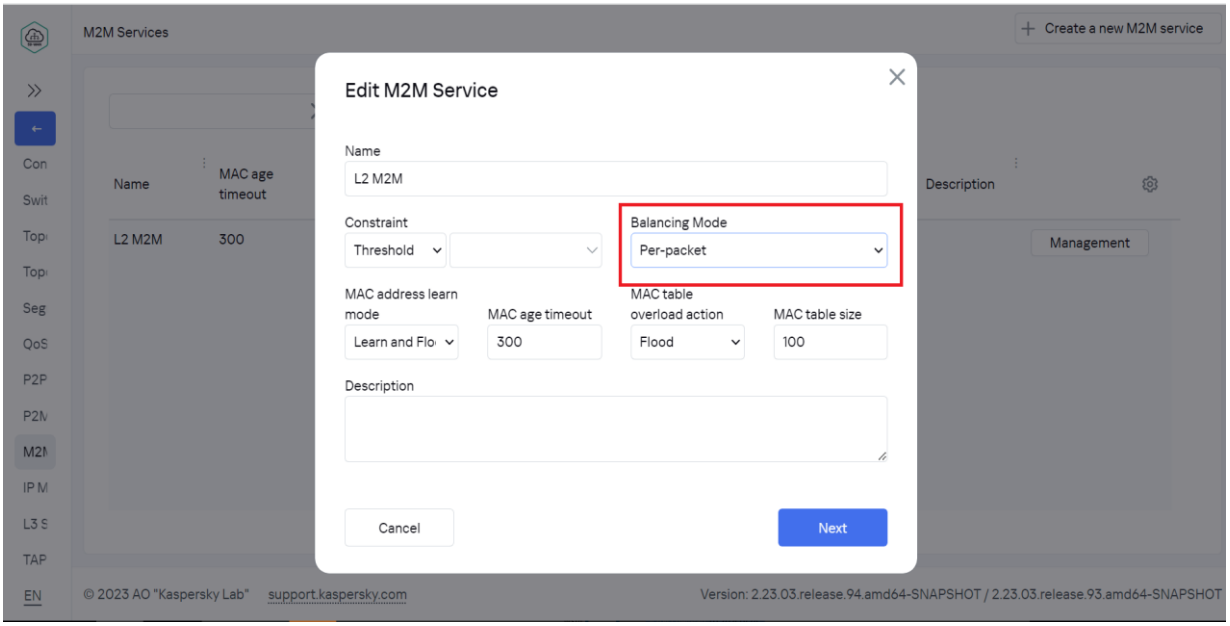
Go to the section with M2M transport services. Select the transport service to edit, click **Management > Edit**.



3.1.3. Включение режим балансировки Per-packet.

Для теста требуется включить режим балансировки Per-packet в связи с тем, что в сценарии для генерации трафика используется iperf, работающий по одному порту. При использовании режима балансировки Per-flow будет задействован только один WAN интерфейс CPE-устройства.

Выбрать Balancing Mode – Per-packet.

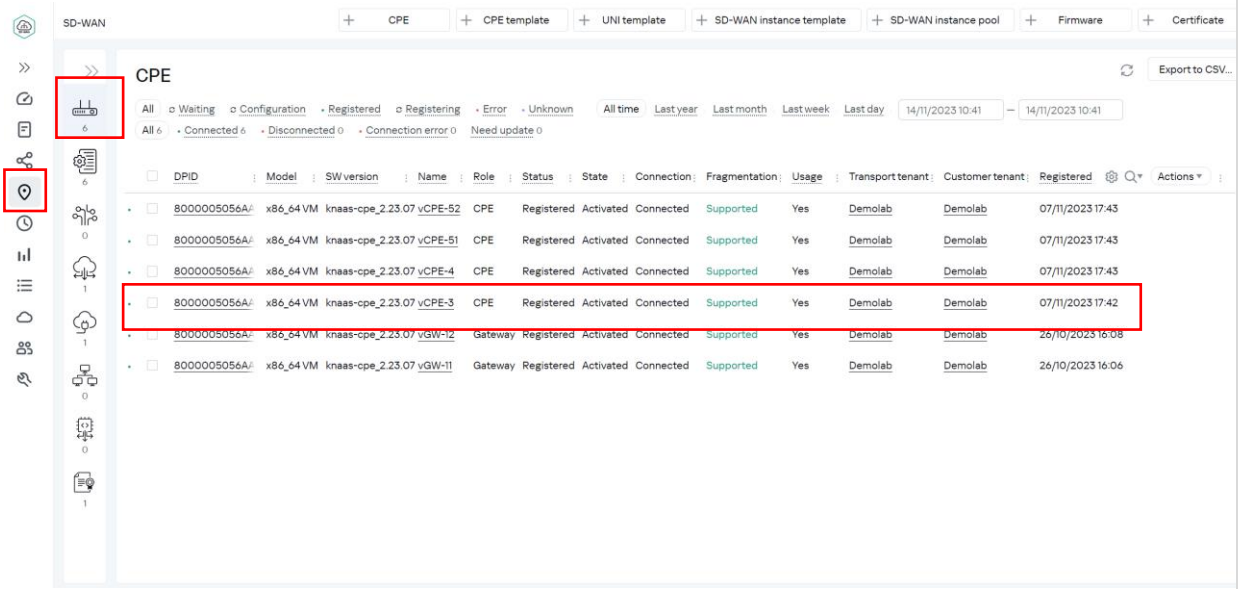


Нажать Next, Next и Save.

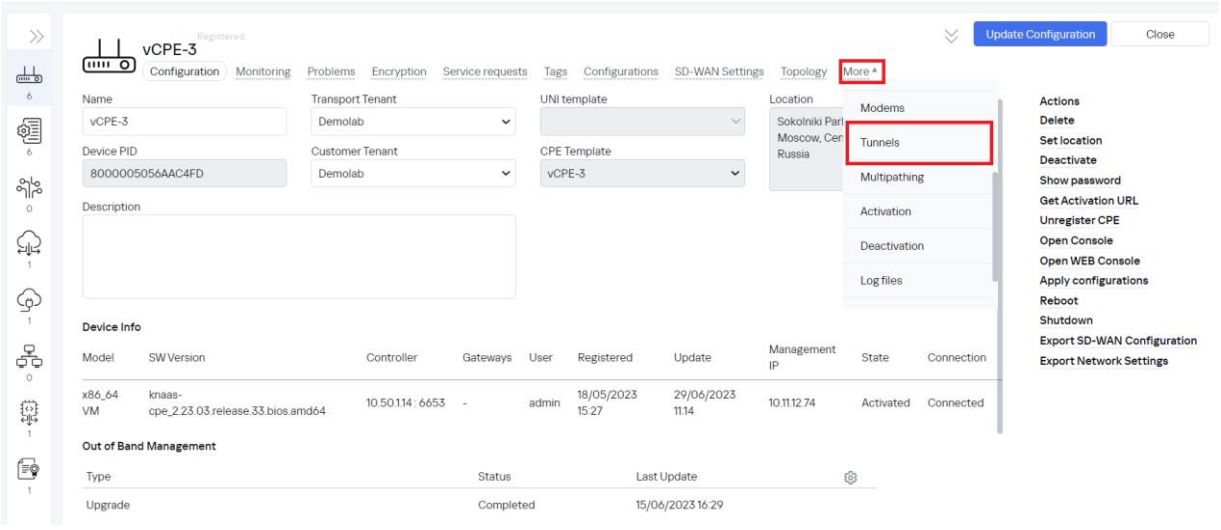
Для получения справочной информации о режимах балансировки обратитесь к Kaspersky SD-WAN Online Help > Создание M2M-сервиса: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/245696.htm

3.1.4. Просмотр построенных туннелей CPE.

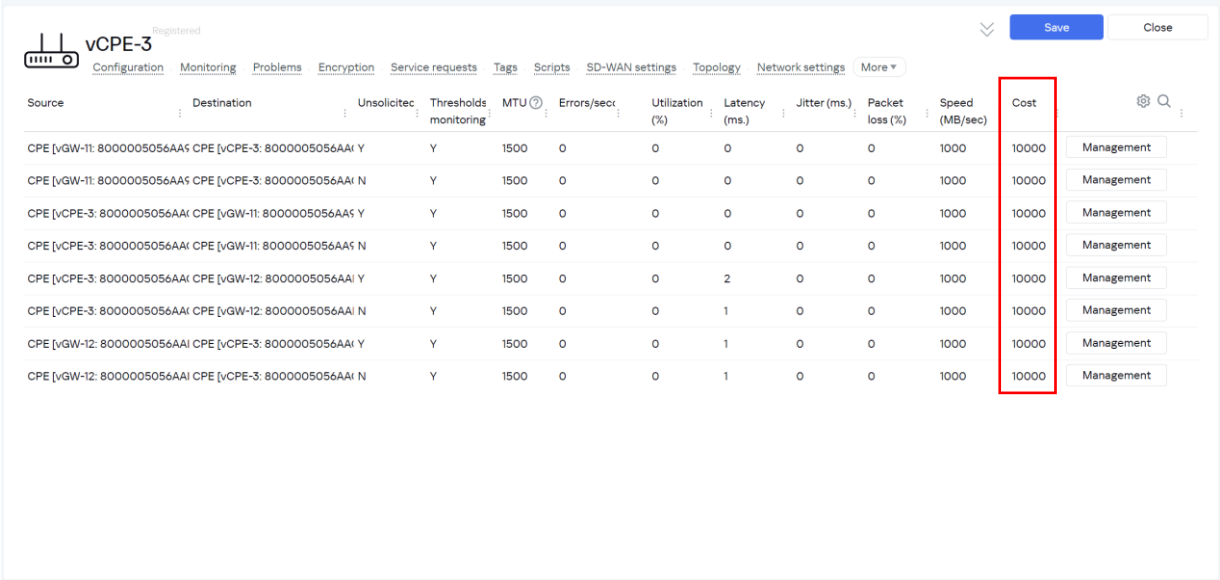
Перейти в меню CPE и выбрать vCPE-3.



Перейти на вкладку Tunnels.



Отобразится список построенных туннелей с vCPE-3. В данном сценарии балансировка будет производится между туннелями с одинаковой стоимостью, без использования multi-weight. Значение стоимости отображается в столбце Cost вкладки Tunnels. Проверить значение стоимости туннелей: для работы балансировки у туннелей должно быть одинаковое значение стоимости.



3.1.5. Генерация тестового трафика.

Для генерации трафика между vCPE-3 и vCPE-4 на рабочих станциях wst3 и wst4 используется iperf.

Запустить сервер iperf на рабочей станции wst4:

[root@wst4]# iperf3 -s

```
[ivpanin@wst4 ~]$ iperf3 -s
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
```

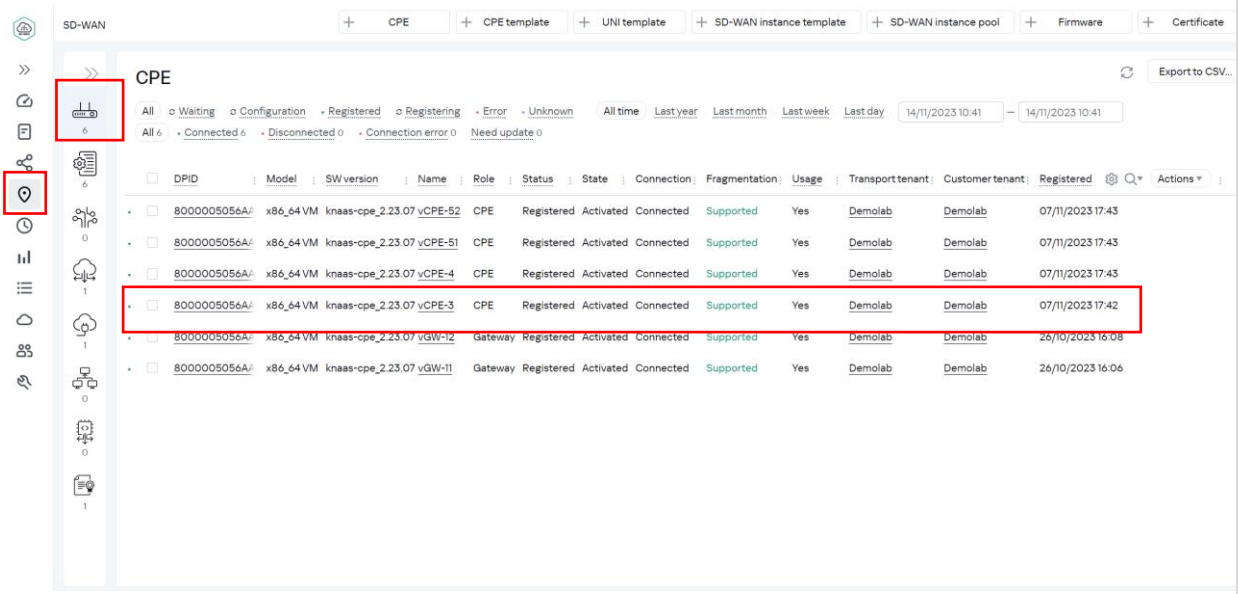
Запустить клиент iperf на рабочей станции wst3:

[root@wst3]# iperf3 -u -t 6000 -c 10.20.4.11

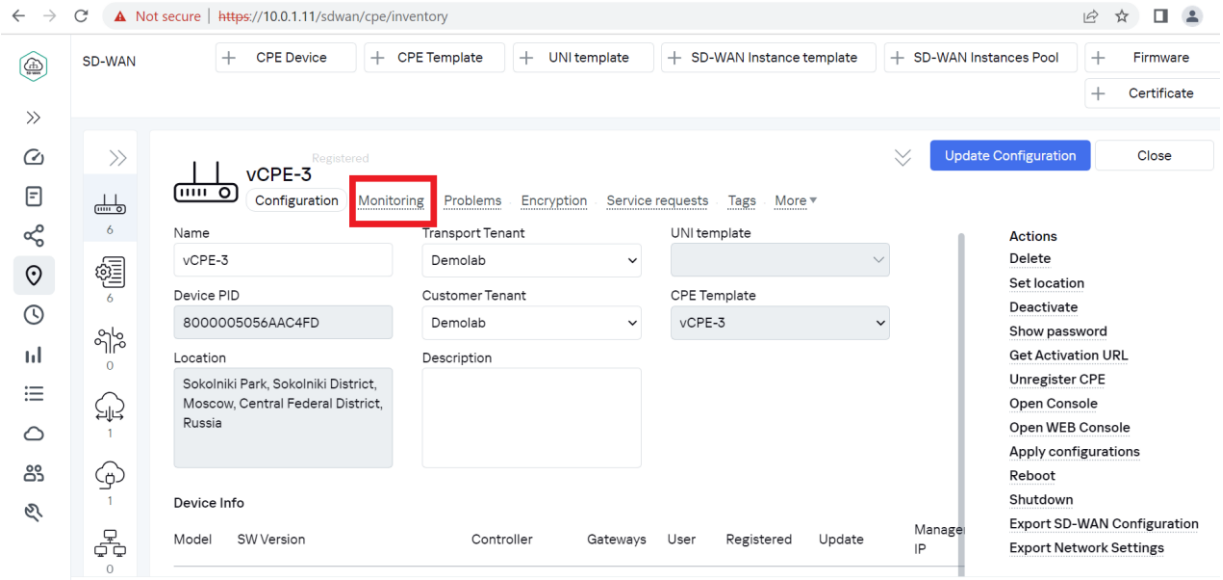
```
[ivpanin@wst3 ~]$ iperf3 -u -t 6000 -c 10.20.4.11
Connecting to host 10.20.4.11, port 5201
[  4] local 10.20.3.11 port 54906 connected to 10.20.4.11 port 5201
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  82
[  4]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  91
[  4]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  90
[  4]   3.00-4.00   sec   129 KBytes  1.05 Mbits/sec  91
```

3.1.6. Проверка балансировки трафика между WAN интерфейсами CPE.

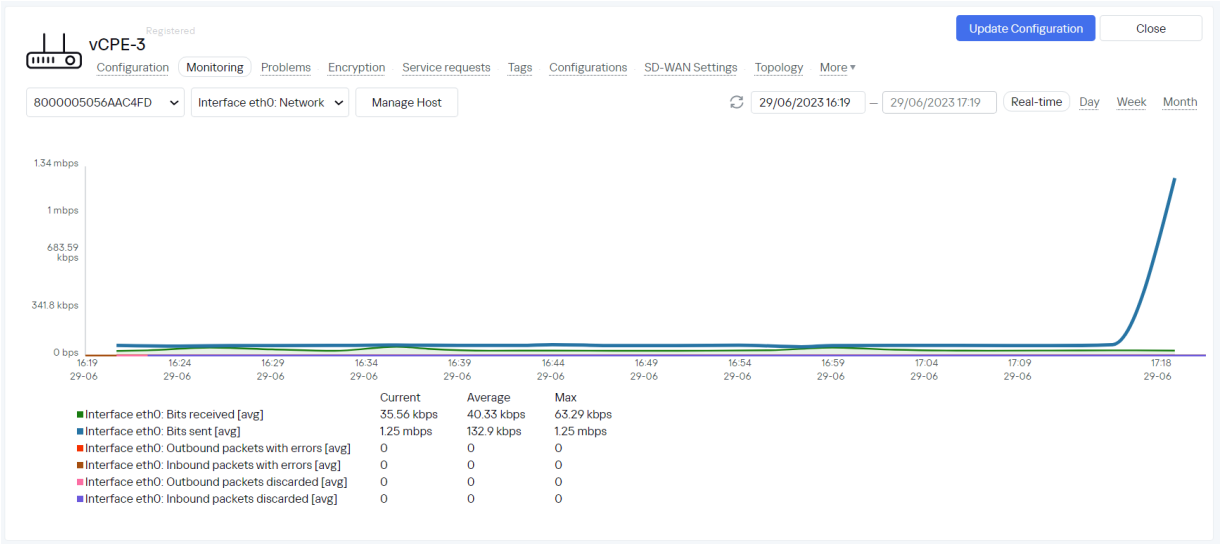
Перейти в меню CPE, открыть vCPE-3.



Открыть вкладку Monitoring.

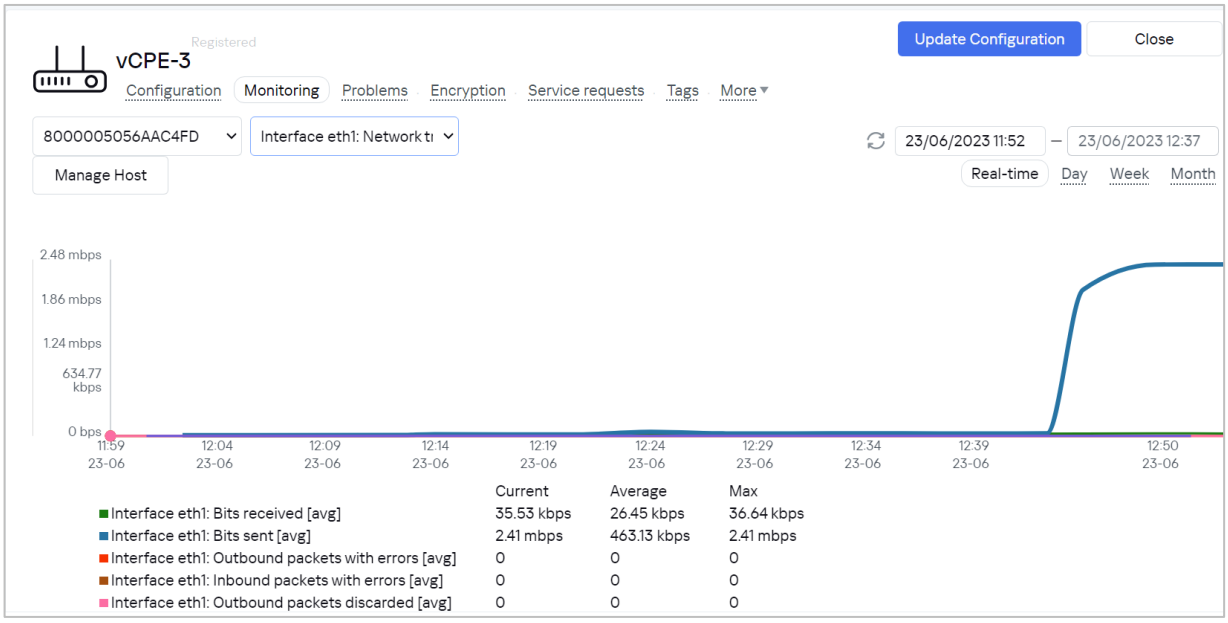


Выбрать интерфейс *eth0* и убедится на графике, что трафик проходит именно через него – всплеск на графике “Interface eth0: Bit sent[avg]”. Для отображения данных необходимо подождать накопления статистики в течении 10 минут.



3.1.7. Проверка прохождения трафика через второй WAN интерфейс CPE.

Выбрать интерфейс *eth1* и убедится на графике, что трафик проходит через данный сетевой интерфейс.



Как видно из графиков в п. 3.1.6 и 3.1.7, в работе участвуют оба WAN интерфейса vCPE-3, и между ними выполняется балансировка трафика.

3.1.8. Возврат настроек после завершения теста

Выполнить п. 3.1.3 и изменить режим балансировки на per-flow.

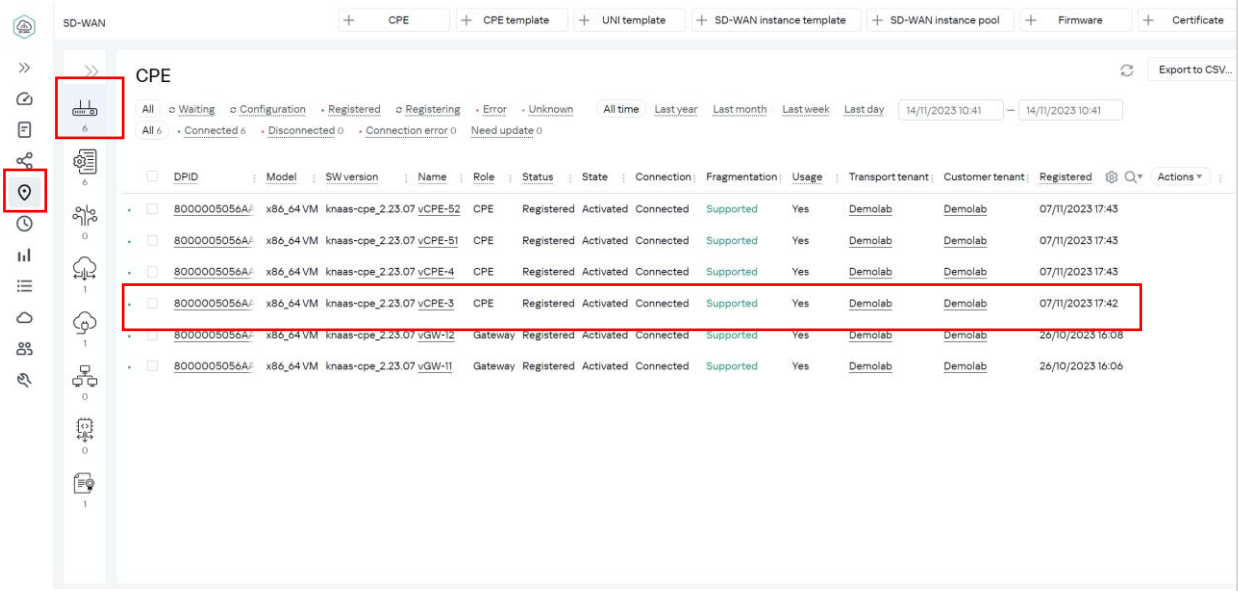
Остановить процессы iperf на wst3 и wst4, запущенные в пункте 3.1.5 (возможно прервать с помощью Ctrl+Z).

3.2. **Резервирование каналов связи в режиме Active/Standby.**

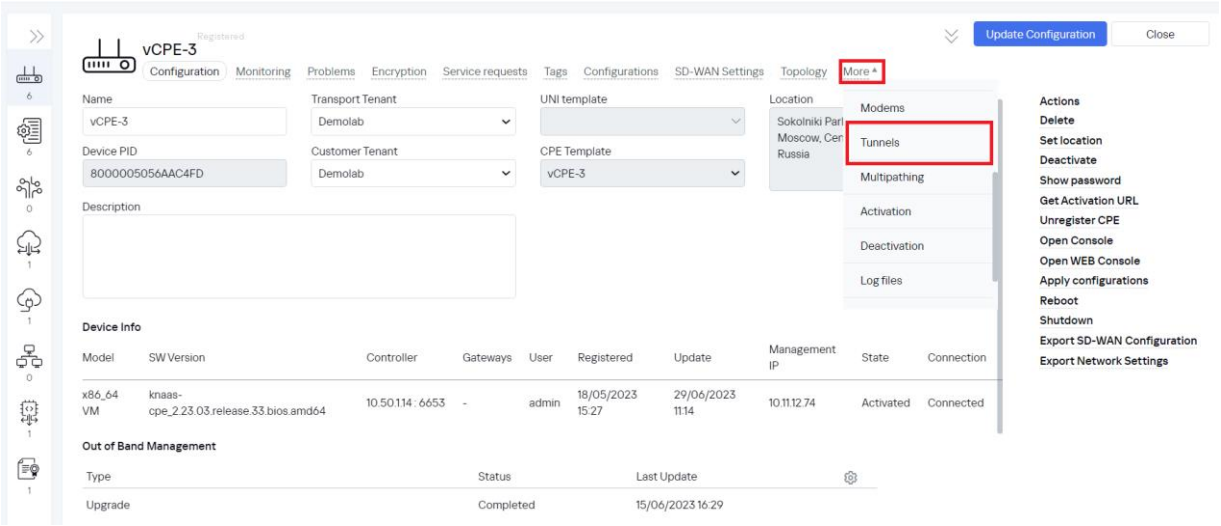
В данном разделе рассматривается сценарий резервирования каналов связи в режиме Active/Standby для устройства vCPE-3. Для приоритезации WAN интерфейса используется параметр Cost, на резервном туннеле значение параметра будет увеличено по сравнению с основным. Генерация трафика на рабочих станциях wst3 и wst4 будет производится с помощью генератора трафика iperf. Для проверки работы резервирования будет использоваться встроенная в решение SD-WAN система мониторинга. Демонстрация работы резервного канала будет производится путем выключения основного WAN-интерфейса CPE.

3.2.1. Отображение списка туннелей vCPE-3 со смежными CPE устройствами.

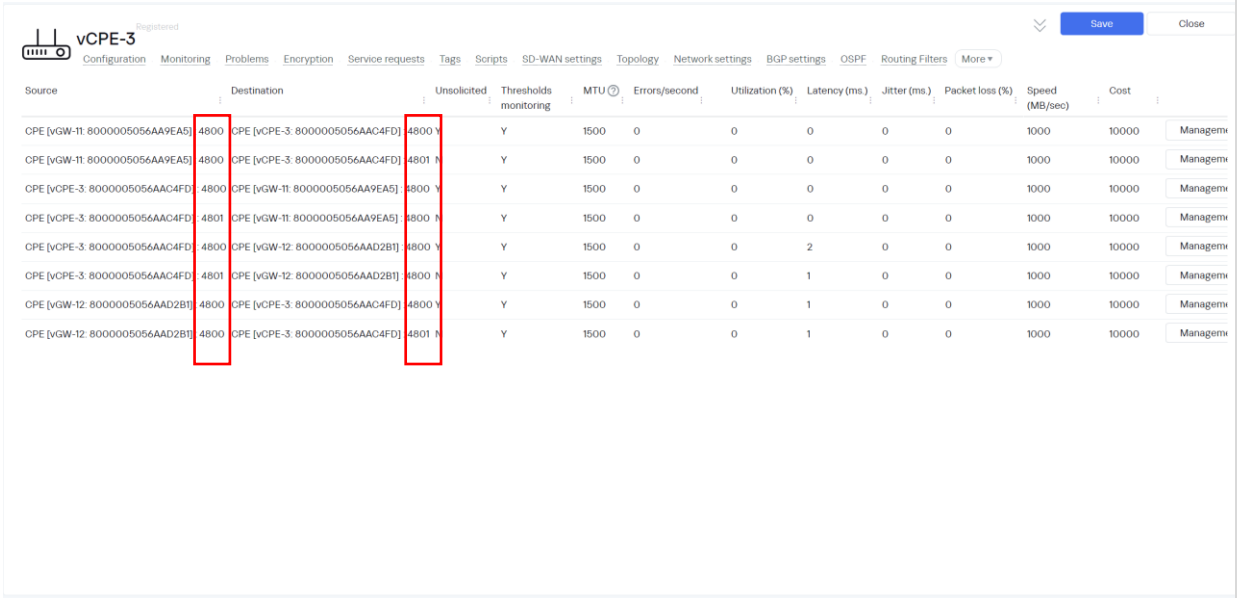
Перейти в меню CPE и выбрать vCPE-3.



Перейти на вкладку Tunnels.



На вкладке Tunnels представлен список построенных туннелей выбранного CPE со смежными CPE устройствами. В столбцах Source и Destination указаны CPE устройства источника и назначения однонаправленного туннеля. Номер порта указывает на номер WAN интерфейса CPE устройства. Номер порта назначается по порядку, начиная с порта 4800, по одному на каждый WAN интерфейс. Порт 4800 означает WAN интерфейс sdwan0 (eth0), порт 4801 означает WAN интерфейс sdwan1 (eth1).



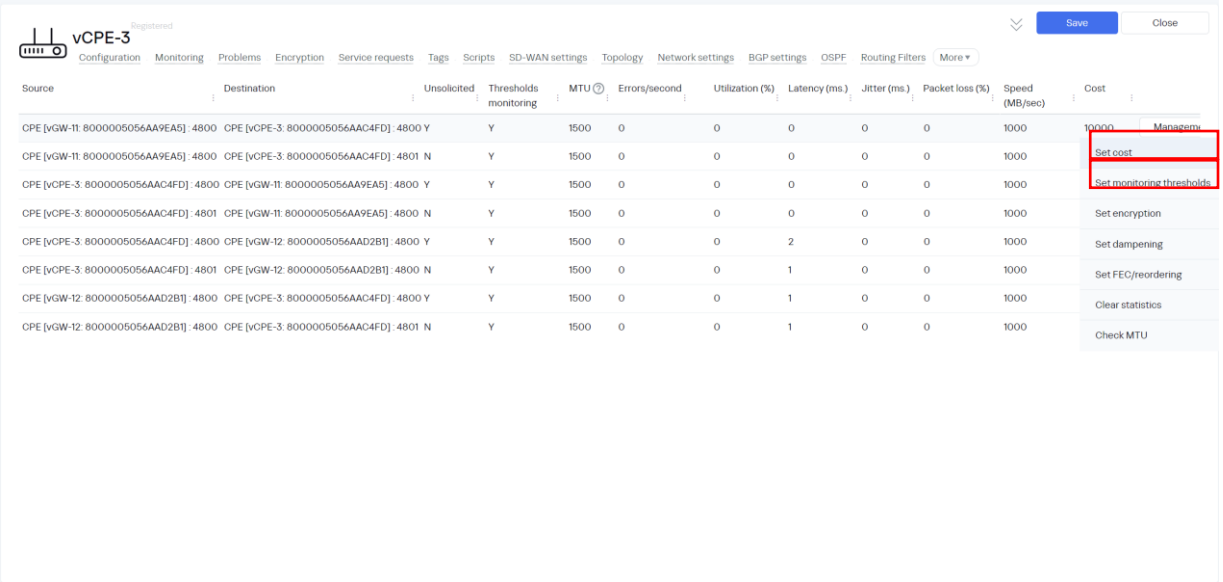
3.2.2. Настройка значения стоимости (Cost) для всех туннелей, построенных через резервный (Standby) WAN интерфейс (sdwan1/eth1) устройства vCPE-3.

В решении SD-WAN топологией по умолчанию является звезда, поэтому трафик между CPE проходит через шлюзы. В данном сценарии будет увеличена стоимость туннелей, проходящих через резервный WAN-интерфейс (sdwan1/eth1) vCPE3, между vCPE-3 и шлюзами vGW-11 / vGW-12.

Найти все туннели между vCPE-3 и vGW-11 / vGW-12, построенные через второй WAN интерфейс vCPE-3, порт 4801:

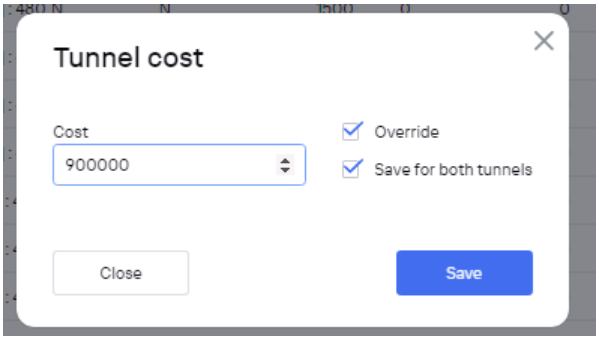
- vCPE-3:4801 <--> vGW-11:4800
- vCPE-3:4801 <--> vGW-12:4800
- vGW-11:4800 <--> vCPE-3:4801
- vGW-12:4800 <--> vCPE-3:4801

Поочередно нажать Management > Set Cost.



Увеличить значение стоимости (по умолчанию 10000):

- В поле Cost задать значение 900000.
- Отметить Override - переопределить значение стоимости.
- Both links - применить настройки к обоим линкам между парой CPE устройств.



kaspersky

3.2.3. Генерация трафика для демонстрации работы Active/Standby на WAN интерфейсах CPE.

Для генерации трафика между vCPE-3 и vCPE-4 на рабочих станциях wst3 и wst4 будет использоваться iperf.

Запустить сервер iperf на wst4:

[root@wst4]# iperf3 -s

```
[ivpanin@wst4 ~]$ iperf3 -s
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
```

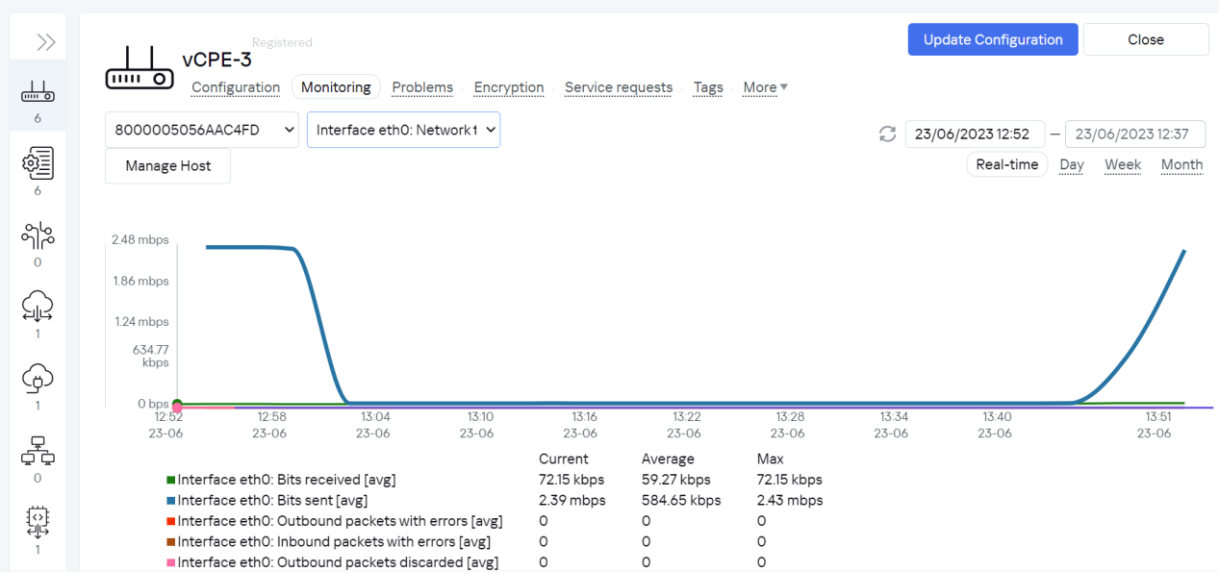
Запустить клиент iperf на wst3:

[root@wst3]# iperf3 -u -t 6000 -c 10.20.4.11

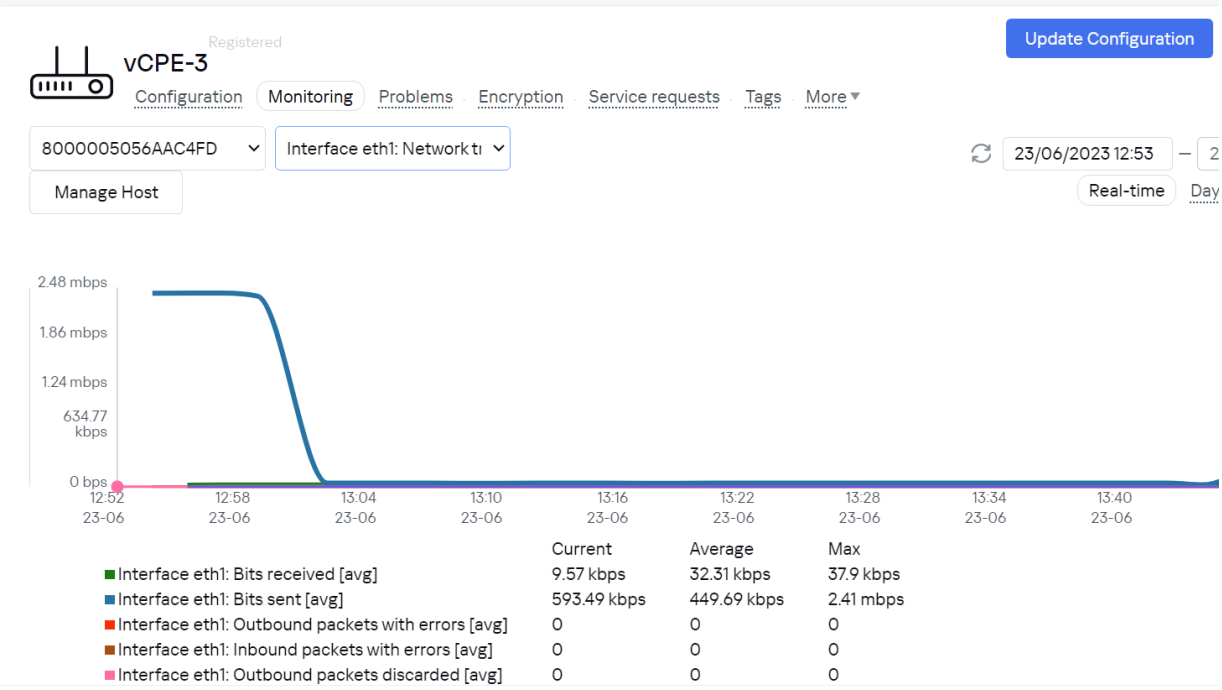
```
[ivpanin@wst3 ~]$ iperf3 -u -t 6000 -c 10.20.4.11
Connecting to host 10.20.4.11, port 5201
[  4] local 10.20.3.11 port 54906 connected to 10.20.4.11 port 5201
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  82
[  4]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  91
[  4]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  90
[  4]   3.00-4.00   sec   129 KBytes  1.05 Mbits/sec  91
```

3.2.4. Проверка статистики трафика на WAN интерфейсах в системе мониторинга.

Перейти в меню CPE, выбрать vCPE-3. Открыть вкладку Monitoring. Выбрать интерфейс *eth0* и убедится на графике, что трафик проходит через него.



Выбрать интерфейс *eth1* и убедиться на графике (кривая “Interface eth0: Bit sent[avg]”) в том, что через данный интерфейс не проходит сетевой трафик.



3.2.5. Эмуляция отказа основного WAN-интерфейса.

Подключиться к маршрутизатору isp и отключить сетевой интерфейс, к которому подключен сетевой интерфейс sdwan0 (eth0) устройства vCPE-3:

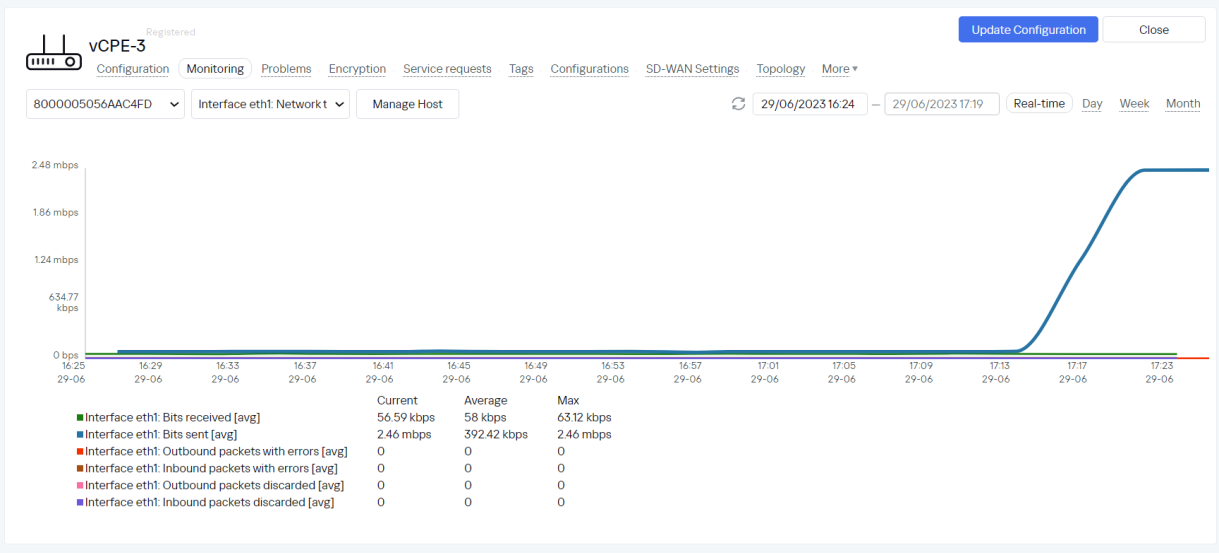
[root@isp]# ifconfig ens161 down

```
root@10.160.0.2's password:
Last failed login: Fri Jun 23 13:54:45 MSK 2023 from 10.160.0.10 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Thu May 18 14:59:19 2023 from 10.160.0.10
[root@isp ~]# ifconfig ens161 down
[root@isp ~]#
```

Из-за особенности работы iperf возможно потребуется перезапустить iperf3 клиент на wst-3: п. 3.2.3

3.2.6. Проверка работы резервирования WAN интерфейсов в системе мониторинга.

Перейти в меню CPE и выбрать vCPE-3. Открыть вкладку Monitoring. Выбрать интерфейс *eth1* и убедиться на графике, что трафик переключился на данный сетевой интерфейс.



3.2.7. Возврат настроек после завершения теста.

Включить сетевой интерфейс на хосте isp, отключенный в п.3.2.5.

[root@isp]# ifconfig ens161 up

Вернуть значение стоимости туннелей, измененное в п. 3.2.2, на значение по умолчанию.
Остановить процессы iperf на wst3 и wst4, запущенные в п. 3.2.3 (возможно прервать с помощью Ctrl+Z).

3.3. **Резервирование каналов связи в широковещательном (broadcast) режиме.**

Kaspersky SD-WAN обеспечивает защиту от перерывов связи с устройствами CPE с помощью одновременного использования доступных каналов передачи данных. Для достижения дополнительной отказоустойчивости поддерживается широковещательный (broadcast) режим балансировки – копии пакетов передаются одновременно во все туннели для исключения потерь.

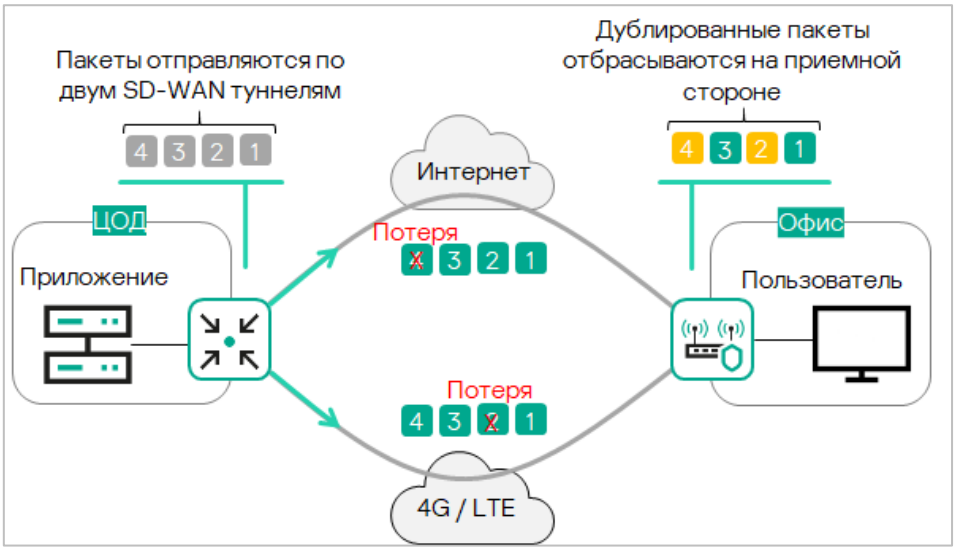


Рисунок 3.3.1 Дублирование пакетов

Для получения дополнительной информации о резервировании каналов связи обратитесь к Kaspersky SD-WAN Online Help > Резервирование каналов передачи данных между устройствами CPE: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/239053.htm

В данном разделе рассматривается сценарий резервирования между туннелями устройства vCPE-3. Для этого будет использоваться режим балансировки пакетов в режиме Broadcast. В данном режиме CPE отправляет копии пакетов одновременно по всем доступным туннелям.

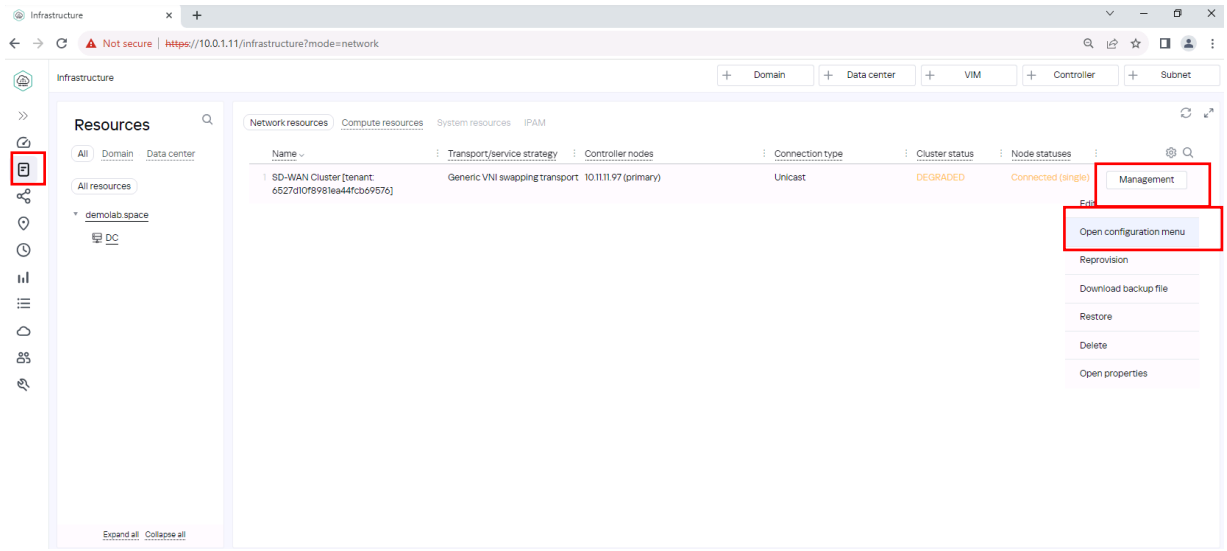
Для демонстрации резервирования трафика между vCPE-3 и srv1 на хостах wst3 и srv1 используется ICMP ping. Для проверки работы механизма дублирования будет использоваться tcpdump на vCPE-3.

3.3.1. Выбор режима балансировки для транспортного сервиса. Настройка режима broadcast для транспортного сервиса.

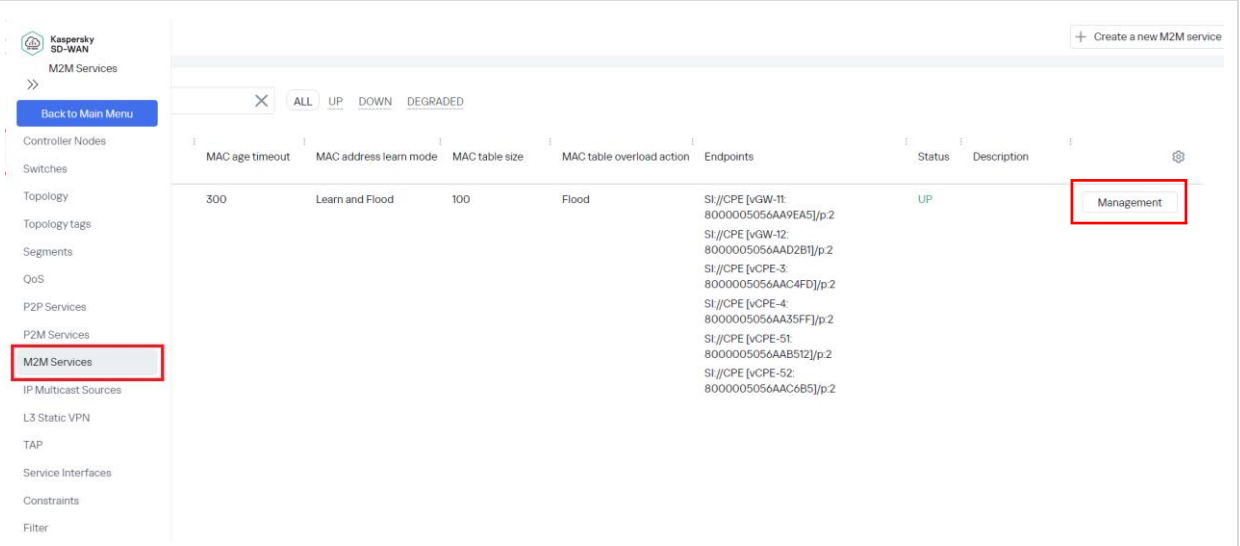
Доступные режимы балансировки:

- Per-flow. Балансировка по потокам (сессиям). При передаче потоки распределяются равномерно по туннелям.
- Per-packet. Балансировка по пакетам. При передаче пакеты распределяются равномерно по туннелям.
- Broadcast. Пакеты передаются одновременно во все туннели для исключения потерь.

Для выбора режима балансировки перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



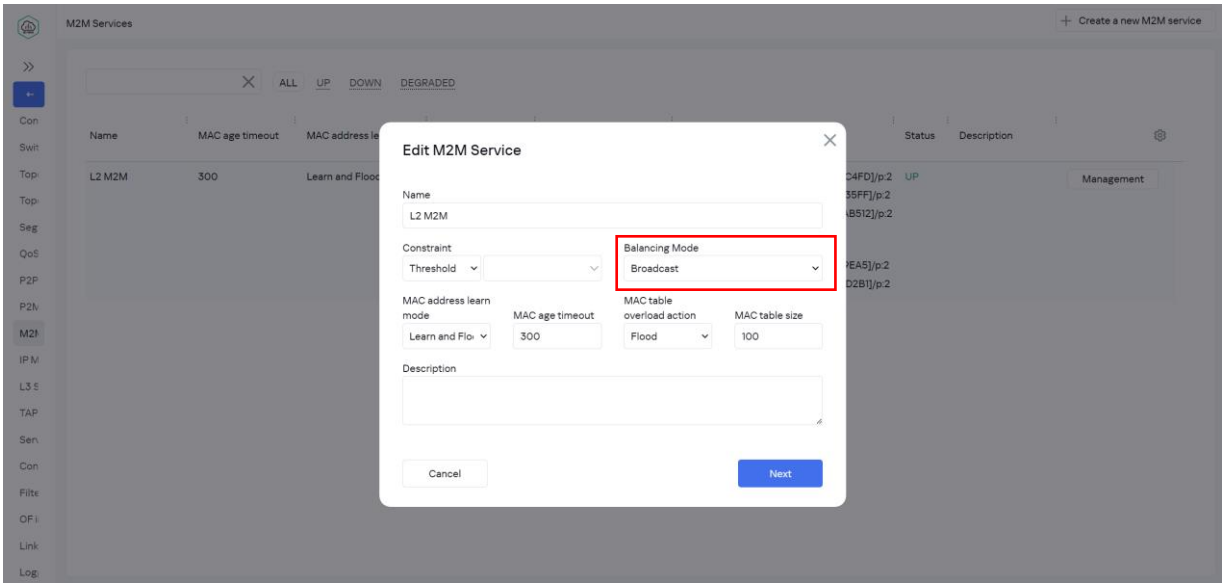
Перейти в раздел с транспортными сервисами M2M. Выбрать транспортный сервис для редактирования, нажать Management > Edit.



Для получения справочной информации о режимах балансировки обратитесь к Kaspersky SD-WAN Online Help > Создание M2M-сервиса: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/245696.htm

3.3.2. Включение режим балансировки Broadcast.

Выбрать Balancing Mode – Broadcast.



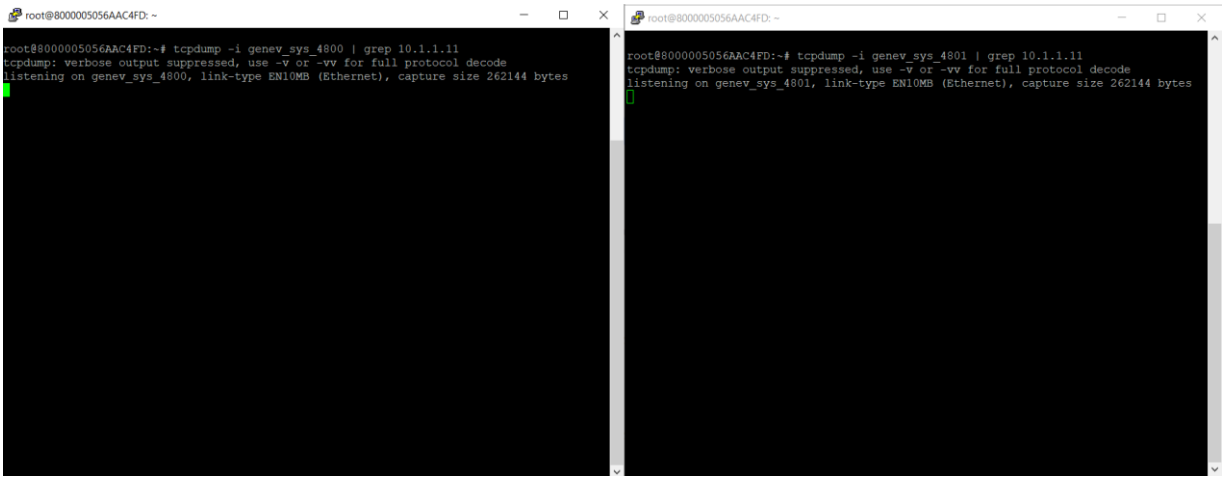
Нажать Next, Next и Save.

3.3.3. Проверка работы режима балансировки broadcast у транспортного сервиса.

Открыть 2 SSH сессии до vCPE-3.
Запустить tcpdump на туннельных интерфейсах: в 1й сессии на genev_sys_4800, во 2й – на genev_sys_4801:

 # tcpdump -i genev_sys_4800 | grep 10.1.1.11
 # tcpdump -i genev_sys_4801 | grep 10.1.1.11

10.1.1.11 – адрес хоста srv1.
genev_sys – туннельные интерфейсы CPE. Номер порта указывает на номер WAN интерфейса CPE устройства. Номер назначается по порядку, начиная с порта 4800, по одному на каждый WAN интерфейс. Порт 4800 означает WAN интерфейс sdwan0 (eth0), порт 4801 означает WAN интерфейс sdwan1 (eth1).



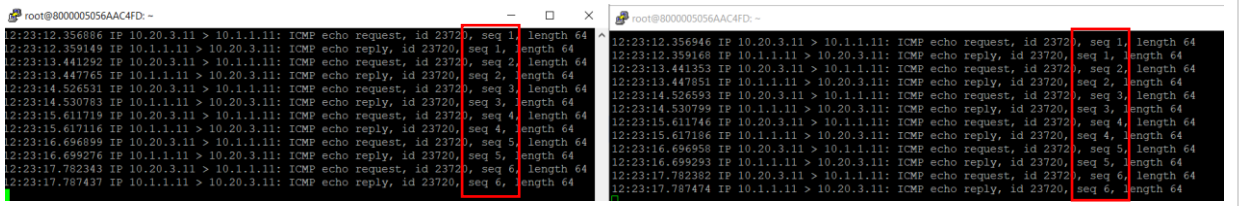
Запустить ICMP ping с wst3 до srv1:

 # ping 10.1.1.11

```
[ivpanin@wst3 ~]$ ping 10.1.1.11

PING 10.1.1.11 (10.1.1.11) 56(84) bytes of data.
64 bytes from 10.1.1.11: icmp_seq=1 ttl=61 time=3.53 ms
64 bytes from 10.1.1.11: icmp_seq=2 ttl=61 time=2.48 ms
64 bytes from 10.1.1.11: icmp_seq=3 ttl=61 time=2.29 ms
64 bytes from 10.1.1.11: icmp_seq=4 ttl=61 time=2.49 ms
64 bytes from 10.1.1.11: icmp_seq=5 ttl=61 time=2.18 ms
64 bytes from 10.1.1.11: icmp_seq=6 ttl=61 time=2.57 ms
```

В выводе tcpdump на vCPE-3 появятся ICMP пакеты. Видно, что на каждый интерфейс была отправлена копия пакетов (у пакетов одинаковый sequence).



3.3.4. Возврат настроек после завершения теста.

Выполнить п. 3.3.2 и изменить режим балансировки на per-flow.

Остановить ICMP ping на wst3, запущенный в пункте 3.3.3 (возможно прервать с помощью Ctrl+Z).

3.4. **Повышение надежности каналов с использованием механизма Forward Error Correction (FEC).**

Функция Forward Error Correction (далее также FEC) позволяет восстанавливать принимаемые данные на устройстве CPE при наличии потерь на каналах передачи данных. Восстановление данных обеспечивается избыточным кодированием потока данных на устройстве, находящемся на передающей стороне.

Передающее устройство CPE кодирует поток выходящих в туннель пакетов трафика с добавлением избыточных пакетов. Степень избыточности можно настроить через параметры контроллера SD-WAN или на отдельном туннеле.

Принимающее устройство CPE буферизует принятые через туннель пакеты трафика и декодирует их с восстановлением потерянных пакетов, если это возможно.

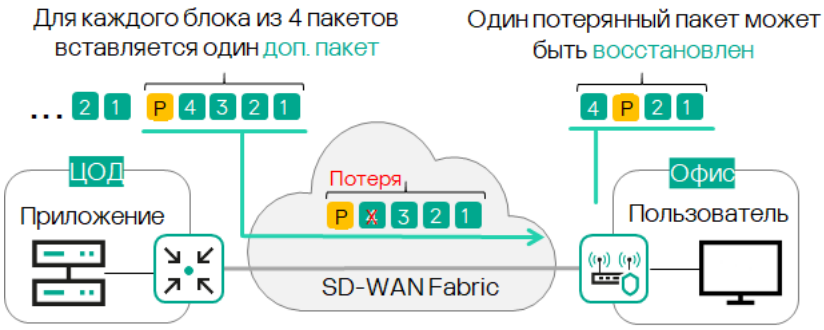


Рис. 3.4.1 Forward Error Correction (FEC)

Использование FEC снижает влияние повышенного показателя потерь пакетов трафика на каналах передачи данных, особенно для UDP-приложений, а также уменьшает количество вызывающих задержки повторных передач пакетов (англ. retransmissions) для TCP-сессий. Рекомендуется использовать FEC на так называемых noisy links (или зашумленных туннелях) для уменьшения коэффициента потери пакетов трафика и увеличения скорости TCP-соединений.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Включение функции Forward Error Correction: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/245033.htm

В данном сценарии рассматривается сценарий с эмуляцией потерь на канале, измерением качества туннелей и включением FEC для восстановления потерянных пакетов. Тестовый трафик будет генерироваться между рабочими станциями wst3 и srv1 с использованием ICMP пинг.

Эмуляция потерь будет проводиться на хосте isp с помощью системы Linux Traffic Control (TC).

3.4.1. Генерация тестового трафика.

Запустить icmp ping с хоста wst3 до srv1:

[root@wst3]# ping 10.1.1.11

```
[ivpanin@wst3 ~]$ ping 10.1.1.11
PING 10.1.1.11 (10.1.1.11) 56(84) bytes of data.
64 bytes from 10.1.1.11: icmp_seq=1 ttl=61 time=11.5 ms
64 bytes from 10.1.1.11: icmp_seq=2 ttl=61 time=7.78 ms
64 bytes from 10.1.1.11: icmp_seq=3 ttl=61 time=5.69 ms
64 bytes from 10.1.1.11: icmp_seq=4 ttl=61 time=3.68 ms
```

3.4.2. Эмуляция потерь пакетов с помощью TC.

Для теста необходимо включить эмуляцию потерь на сетевом интерфейсе хоста isp, к которому подключен sdwan0 (eth0) интерфейс vCPE-3.

Подключиться к хосту isp и выполнить команду:

[root@isp]# tc qdisc add dev ens161 root netem delay 1ms 0ms limit 1250000 loss 5%

Данная команда создает 5% потерь (packet loss). Параметр delay настраивает задержку в 1ms с разбросом в 0ms, limit – выделяет буфер в 1250000 байт для обработки данных TC
Проверить примененные настройки с помощью следующей команды:

[root@isp]# tc qdisc show

```
[root@isp ~]# tc qdisc add dev ens161 root netem delay 1ms 0ms limit 1250000 loss 10%
[root@isp ~]# tc qdisc show
qdisc noqueue 0: dev lo root refcnt 2
qdisc netem 8005: dev ens161 root refcnt 2 limit 1250000 delay 1.0ms loss 10%
qdisc pfifo_fast 0: dev ens192 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens193 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens224 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens225 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens256 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens257 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
[root@isp ~]#
```

3.4.3. Проверка на рабочей станции wst3 потерь в статистике ping.

Примечание: по умолчанию режим балансировки per-flow (если он не был изменен на per-packet в 3.1), поэтому поток может пойти через другой интерфейс, и эмуляция потерь не применится на интерфейс, через который будет проходить поток трафика. В данном сценарии должен использоваться режим per-flow.

Как видно ниже по ICMP sequence number, присутствуют потери пакетов, видны пропущенные ответы (пропущены sequence 281, 290, 293).

```
64 bytes from 10.1.1.11: icmp_seq=275 ttl=61 time=29.8 ms
64 bytes from 10.1.1.11: icmp_seq=276 ttl=61 time=7.69 ms
64 bytes from 10.1.1.11: icmp_seq=277 ttl=61 time=5.76 ms
64 bytes from 10.1.1.11: icmp_seq=278 ttl=61 time=43.8 ms
64 bytes from 10.1.1.11: icmp_seq=279 ttl=61 time=21.8 ms
64 bytes from 10.1.1.11: icmp_seq=280 ttl=61 time=79.7 ms
64 bytes from 10.1.1.11: icmp_seq=282 ttl=61 time=17.6 ms
64 bytes from 10.1.1.11: icmp_seq=283 ttl=61 time=15.8 ms
64 bytes from 10.1.1.11: icmp_seq=286 ttl=61 time=14.6 ms
64 bytes from 10.1.1.11: icmp_seq=287 ttl=61 time=12.8 ms
64 bytes from 10.1.1.11: icmp_seq=288 ttl=61 time=10.7 ms
64 bytes from 10.1.1.11: icmp_seq=289 ttl=61 time=28.7 ms
64 bytes from 10.1.1.11: icmp_seq=291 ttl=61 time=66.7 ms
64 bytes from 10.1.1.11: icmp_seq=292 ttl=61 time=4.65 ms
64 bytes from 10.1.1.11: icmp_seq=294 ttl=61 time=5.94 ms
64 bytes from 10.1.1.11: icmp_seq=295 ttl=61 time=41.5 ms
64 bytes from 10.1.1.11: icmp_seq=296 ttl=61 time=79.7 ms
64 bytes from 10.1.1.11: icmp_seq=297 ttl=61 time=17.7 ms
64 bytes from 10.1.1.11: icmp_seq=298 ttl=61 time=15.7 ms
64 bytes from 10.1.1.11: icmp_seq=300 ttl=61 time=13.8 ms
64 bytes from 10.1.1.11: icmp_seq=301 ttl=61 time=32.6 ms
64 bytes from 10.1.1.11: icmp_seq=302 ttl=61 time=11.7 ms
```

Если в статистике не будет видно потерь, то значит, что трафик идет через интерфейс, где не применена эмуляция потерь и необходимо на хосте isp применить эмуляцию на другой интерфейс:

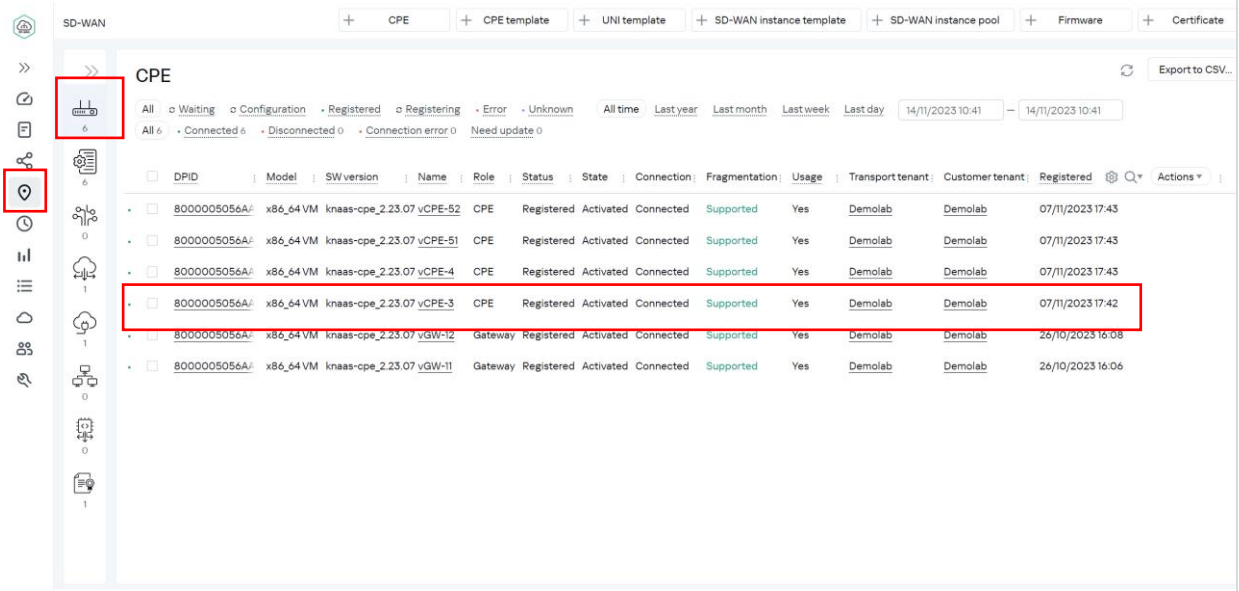
[root@isp]# tc qdisc add dev ens193 root netem delay 1ms 0ms limit 1250000 loss 5%

и снять задержку с первого сетевого интерфейса:

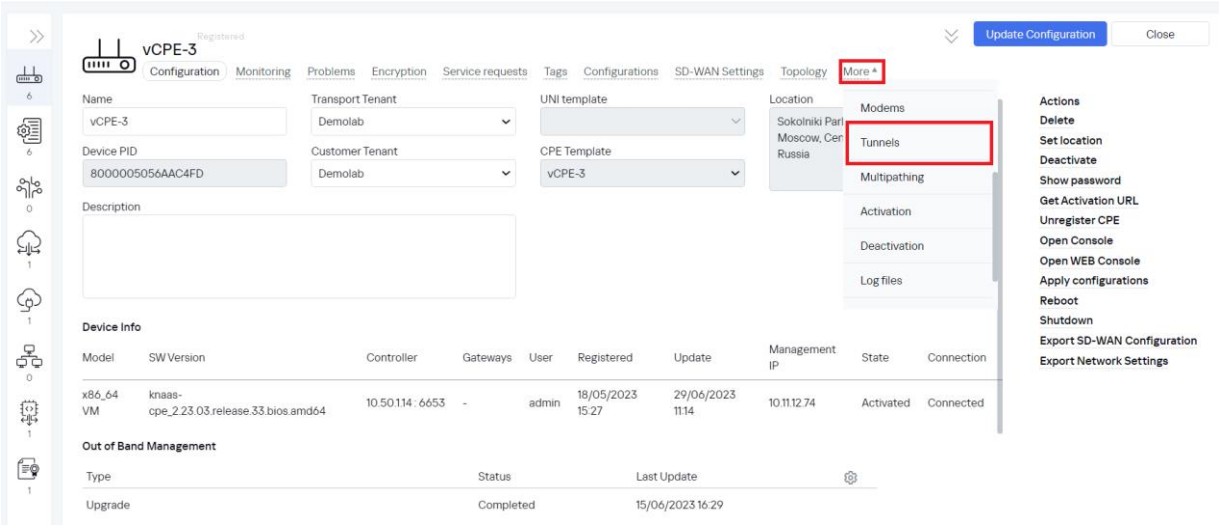
[root@isp]# tc qdisc del dev ens161 root

3.4.4. Включение мониторинга потерь пакетов на туннелях.

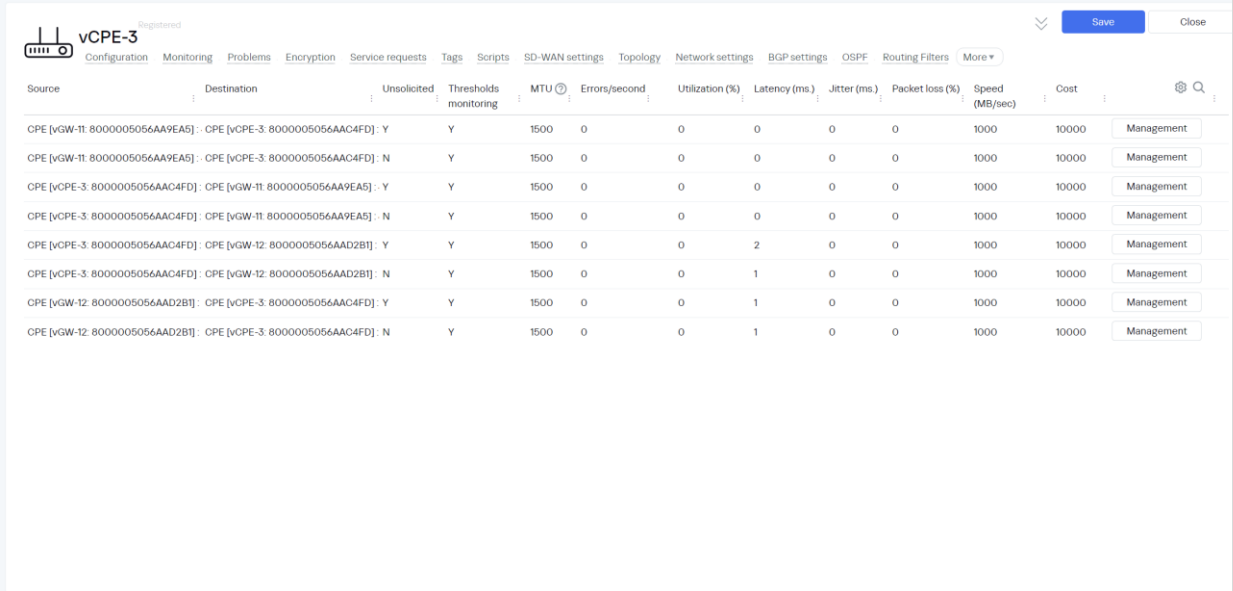
Перейти в меню CPE и выбрать vCPE-3.



Перейти на вкладку Tunnels.



Отобразится список построенных туннелей с vCPE-3.

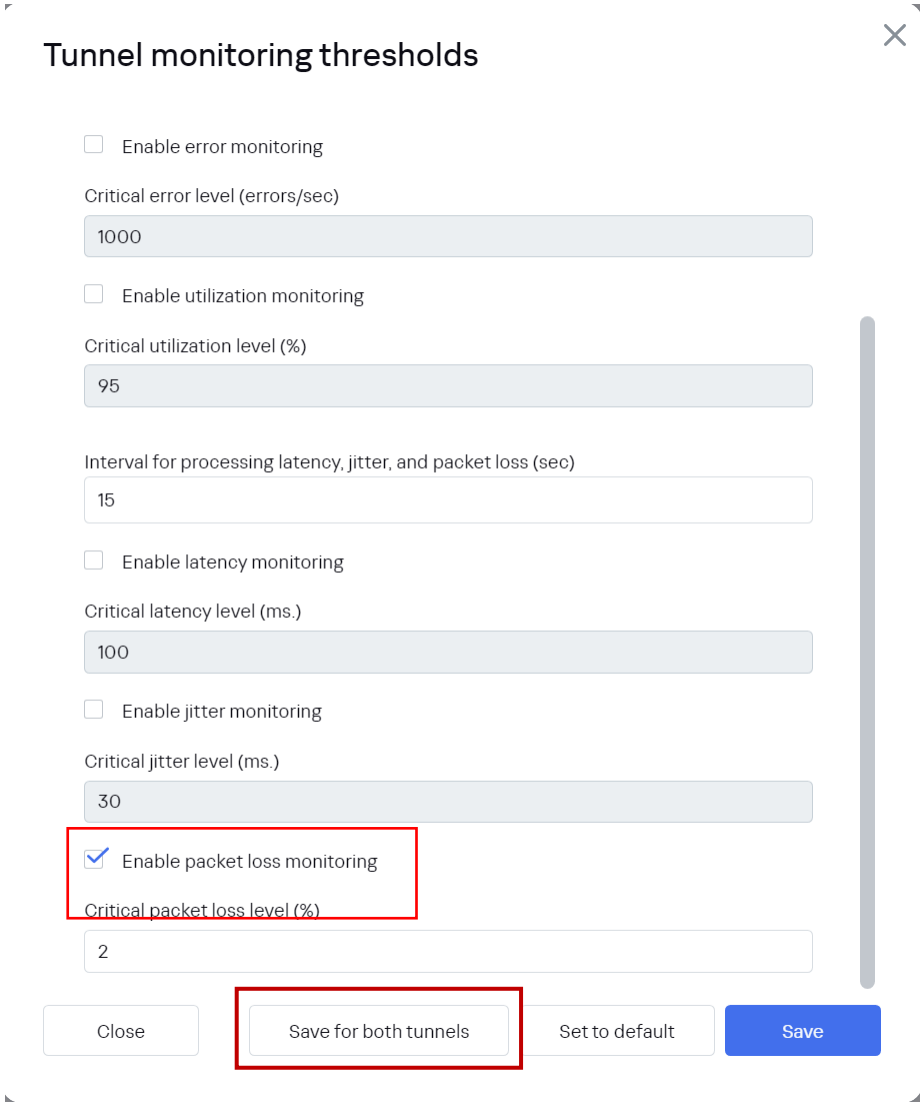


Нажать Management > Set monitoring thresholds.

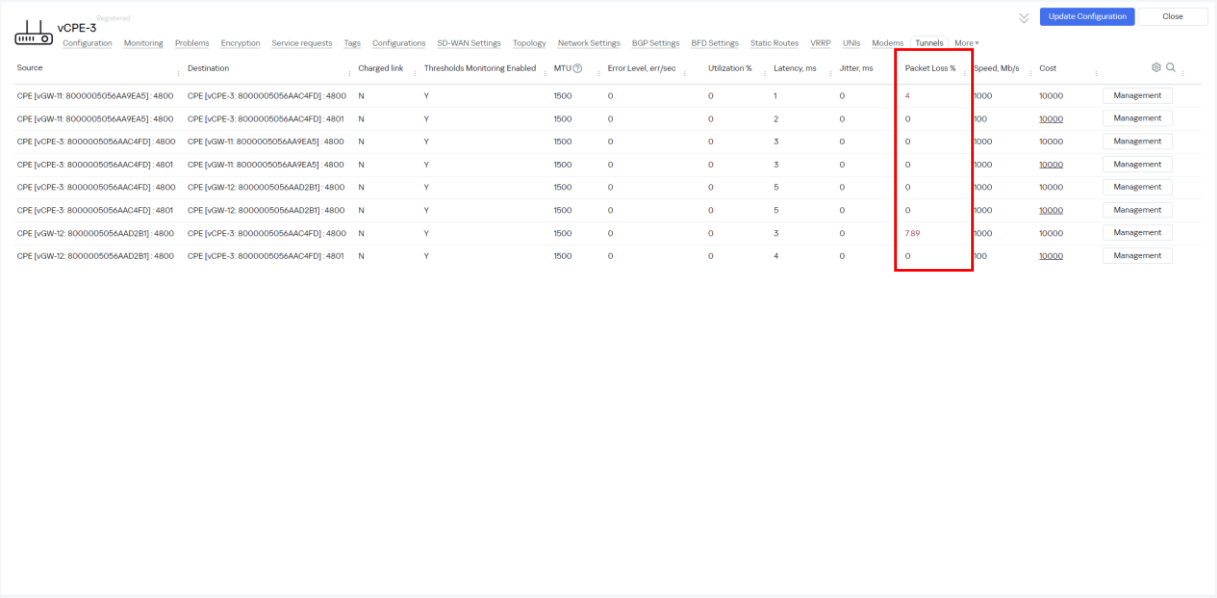
Включить:

- Enable tunnel thresholds monitoring
- Enable packet loss monitoring > Critical packet loss level – 2%.

Нажать Save for both tunnels – сохранение параметров мониторинга туннелей в оба направления.



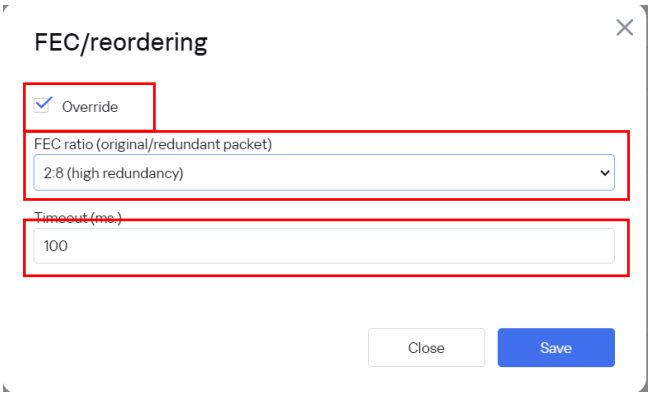
Повторить настройки для всех туннелей.
После применения настроек отобразится статистика потерь на туннелях. Значения измеренных параметров, не удовлетворяющих порогам, заданных ранее, будут выделены красным цветом. Т.к. задержка эмулировалась в сторону интерфейс sdwan0(eth0) vCPE-3, то packet loss наблюдается на соответствующих туннелях до vGW-11 и vGW-12, проходящих через данный интерфейс.



3.4.5. Включение FEC.

Для туннелей необходимо настроить Forward Error Correction.

Поочередно для каждого туннеля выбрать Management > Set FEC/Reordering.
Отметить Override и задать FEC ratio 2:8.
Задать Timeout – 100.
Нажать Save.



3.4.6. Проверка работы FEC в статистике ping.

Проверить на хосте wst3, что в статистике ping пропали пропущенные ICMP ответы. В статистике видно, что все ICMP пакеты успешно прошли: по номерам sequence не видно пропусков. Пакеты успешно восстанавливаются с помощью избыточного кодирования.

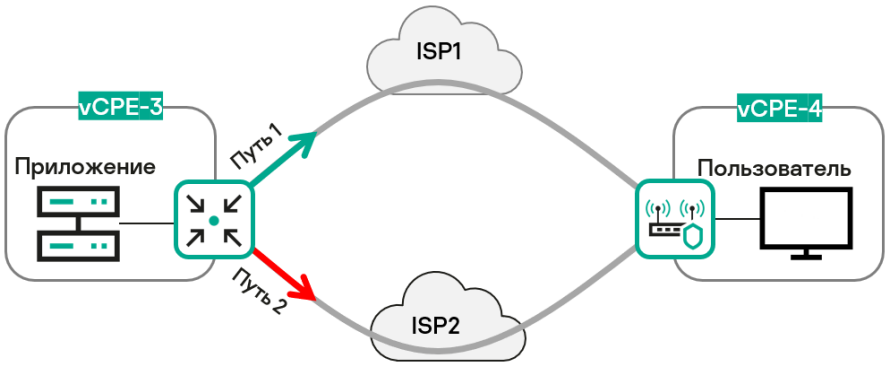
```
64 bytes from 10.1.1.11: icmp_seq=554 ttl=61 time=16.8 ms
64 bytes from 10.1.1.11: icmp_seq=555 ttl=61 time=15.6 ms
64 bytes from 10.1.1.11: icmp_seq=556 ttl=61 time=73.6 ms
64 bytes from 10.1.1.11: icmp_seq=557 ttl=61 time=11.7 ms
64 bytes from 10.1.1.11: icmp_seq=558 ttl=61 time=29.8 ms
64 bytes from 10.1.1.11: icmp_seq=559 ttl=61 time=27.6 ms
64 bytes from 10.1.1.11: icmp_seq=560 ttl=61 time=5.72 ms
64 bytes from 10.1.1.11: icmp_seq=561 ttl=61 time=3.68 ms
64 bytes from 10.1.1.11: icmp_seq=562 ttl=61 time=21.8 ms
64 bytes from 10.1.1.11: icmp_seq=563 ttl=61 time=19.7 ms
64 bytes from 10.1.1.11: icmp_seq=564 ttl=61 time=16.4 ms
64 bytes from 10.1.1.11: icmp_seq=565 ttl=61 time=6.86 ms
64 bytes from 10.1.1.11: icmp_seq=566 ttl=61 time=12.3 ms
64 bytes from 10.1.1.11: icmp_seq=567 ttl=61 time=10.7 ms
64 bytes from 10.1.1.11: icmp_seq=568 ttl=61 time=8.50 ms
64 bytes from 10.1.1.11: icmp_seq=569 ttl=61 time=7.38 ms
64 bytes from 10.1.1.11: icmp_seq=570 ttl=61 time=5.83 ms
64 bytes from 10.1.1.11: icmp_seq=571 ttl=61 time=3.76 ms
64 bytes from 10.1.1.11: icmp_seq=572 ttl=61 time=21.8 ms
64 bytes from 10.1.1.11: icmp_seq=573 ttl=61 time=19.6 ms
64 bytes from 10.1.1.11: icmp_seq=574 ttl=61 time=17.8 ms
64 bytes from 10.1.1.11: icmp_seq=575 ttl=61 time=75.7 ms
```

3.4.7. Возврат настроек после завершения теста.
Выполнить п. 3.4.4 и выключить мониторинг потерь пакетов для туннелей.
Выполнить п. 3.4.5 и выключить FEC для туннелей.
Остановить ICMP ping на wst3, запущенный в пункте 3.4.1 (возможно прервать с помощью Ctrl+Z).
Выключить эмуляцию задержек и джиттера на хосте isp.

[root@isp]# tc qdisc del dev ens161 root netem
[root@isp]# tc qdisc del dev ens193 root netem

3.5. **Мониторинг качества туннелей (Jitter, Latency, Packet Loss) и управление трафиком в соответствии с заданным SLA.**

Решение SD-WAN позволяет производить измерения параметров прохождения пакетов через туннели (джиттер, задержка, потери пакетов) и изменять пути прохождения трафика в зависимости от заданных параметров, например, чтобы обеспечить минимальную задержку. Измерения параметров туннеля производится с использованием дополнительных полей Type-Length Value (TLV) внутри заголовков GENEVE.



	Джиттер	Потери пакетов	Задержка
Путь 1	71 ms	0 %	297
Путь 2	4 ms	2 %	15

Рис. 3.5.1 Мониторинг качества туннелей

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Включение мониторинга на туннеле:

https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/244988.htm

Ниже рассматривается сценарий с измерением задержки и джиттера на туннелях, заданием ограничений и перенаправление трафика на туннели, которые удовлетворяют ограничениям на задержку и джиттер. Тестовый трафик будет генерироваться между рабочими станциями wst3 и wst4 с использованием iperf, также в статистике iperf будет проверяться статистика джиттера.

Эмуляция задержек и джиттера будет проводиться на хосте isp с помощью системы Linux Traffic Control.

Будут созданы ограничения для транспортного сервиса с целью исключения туннелей, не удовлетворяющих заданным параметрам джиттера и задержек.

Для корректной работы мониторинга задержек все устройства CPE и шлюзы должны иметь доступ к NTP серверам и время на устройствах должно быть синхронизировано.

3.5.1. Генерация тестового трафика.

Запустить сервер iperf на хосте wst4:

[root@wst4]# iperf3 -s | grep ms

```
[ivpanin@wst4 ~]$
[ivpanin@wst4 ~]$ iperf3 -s | grep ms
```

Запустить клиент iperf на хосте wst3:

[root@wst3]# iperf3 -u -t 6000 -c 10.20.4.11

```
[ivpanin@wst3 ~]$ iperf3 -u -t 6000 -c 10.20.4.11
Connecting to host 10.20.4.11, port 5201
[  4] local 10.20.3.11 port 43730 connected to 10.20.4.11 port 5201
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  82
[  4]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  91
[  4]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  90
```

В случае успешного соединения, в статистике на wst4 появится количество принятых пакетов.

```
[ivpanin@wst4 ~]$ iperf3 -s | grep ms
[ ID] Interval           Transfer     Bandwidth       Jitter
[  5]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  0.128 ms
[  5]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  0.133 ms
[  5]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  0.117 ms
[  5]   3.00-4.00   sec   129 KBytes  1.05 Mbits/sec  0.162 ms
[  5]   4.00-5.00   sec   127 KBytes  1.04 Mbits/sec  0.141 ms
[  5]   5.00-6.00   sec   129 KBytes  1.05 Mbits/sec  0.149 ms
[  5]   6.00-7.00   sec   127 KBytes  1.04 Mbits/sec  0.161 ms
```

3.5.2. Эмуляция задержки и джиттера с помощью TC.

Для теста необходимо включить эмуляцию задержки и джиттера на сетевом интерфейсе хоста isp, к которому подключен sdwan0 (eth0) интерфейс vCPE-3.

Подключиться к хосту isp и выполнить команду:

[root@isp]# tc qdisc add dev ens193 root netem delay 300ms 100ms

Данная команда создает задержку (delay / latency) в 300ms с разбросом (jitter) в 100ms.

Проверить примененные настройки с помощью следующей команды:

[root@isp]# tc qdisc show

```
[root@isp ~]#
[root@isp ~]# tc qdisc add dev ens193 root netem delay 300ms 100ms
[root@isp ~]# tc qdisc show
qdisc noqueue 0: dev lo root refcnt 2
qdisc pfifo_fast 0: dev ens161 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens192 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc netem 8001: dev ens193 root refcnt 2 limit 1000 delay 300.0ms  100.0ms
qdisc pfifo_fast 0: dev ens224 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens225 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens256 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
qdisc pfifo_fast 0: dev ens257 root refcnt 2 bands 3 priomap  1 2 2 2 1 2 0 0 1 1 1 1 1 1 1 1
[root@isp ~]#
```

3.5.3. Проверка на рабочей станции wst4 наличие джиттера в статистике iperf.

Примечание: по умолчанию режим балансировки per-flow (если он не был изменен на per-packet в 3.1), поэтому поток может пойти через другой интерфейс, и джиттера может не быть.

```
[  5] 379.00-380.00 sec   129 KBytes  1.05 Mbits/sec  0.156 ms  21/91 (23%)
[  5] 380.00-381.00 sec   127 KBytes  1.04 Mbits/sec  0.136 ms  8/90 (8.9%)
[  5] 381.00-382.00 sec   129 KBytes  1.05 Mbits/sec  0.166 ms  19/91 (21%)
[  5] 382.00-383.00 sec   127 KBytes  1.04 Mbits/sec  0.132 ms  26/90 (29%)
[  5] 383.00-384.00 sec   129 KBytes  1.05 Mbits/sec  0.147 ms  16/91 (18%)
[  5] 384.00-385.00 sec   127 KBytes  1.04 Mbits/sec  0.128 ms  8/90 (8.9%)
[  5] 385.00-386.00 sec  86.3 KBytes   706 Kbits/sec  43.065 ms  34/63 (54%)
[  5] 386.00-387.00 sec   174 KBytes  1.43 Mbits/sec  32.205 ms  63/95 (66%)
[  5] 387.00-388.00 sec   188 KBytes  1.54 Mbits/sec  45.508 ms  50/87 (57%)
[  5] 388.00-389.00 sec   160 KBytes  1.31 Mbits/sec  29.382 ms  64/97 (66%)
[  5] 389.00-390.00 sec   170 KBytes  1.39 Mbits/sec  39.571 ms  56/87 (64%)
[  5] 390.00-391.00 sec   160 KBytes  1.31 Mbits/sec  24.279 ms  57/96 (59%)
[  5] 391.00-392.00 sec   163 KBytes  1.33 Mbits/sec  33.819 ms  52/82 (63%)
[  5] 392.00-393.00 sec   165 KBytes  1.36 Mbits/sec  44.783 ms  66/97 (68%)
[  5] 393.00-394.00 sec   148 KBytes  1.22 Mbits/sec  41.437 ms  55/92 (60%)
[  5] 394.00-395.00 sec   129 KBytes  1.05 Mbits/sec  32.055 ms  45/81 (56%)
[  5] 395.00-396.00 sec   156 KBytes  1.27 Mbits/sec  40.827 ms  57/94 (61%)
```

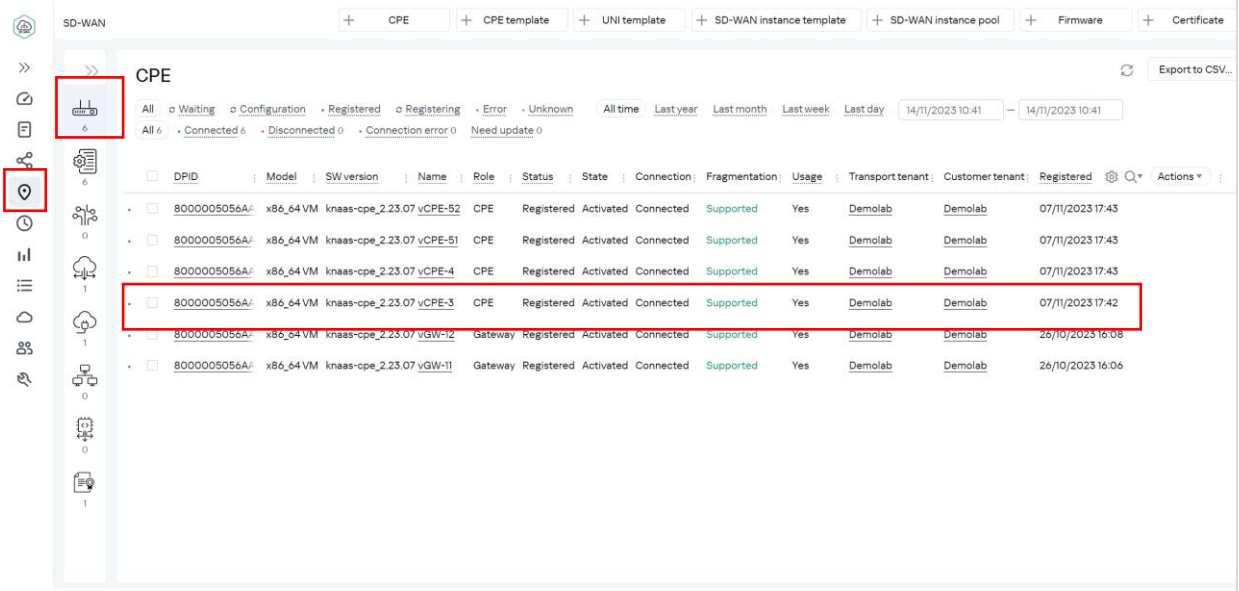
Если в статистике не будет видно задержку, то необходимо на хосте isp применить эмуляцию на другой интерфейс:

[root@isp]# tc qdisc add dev ens161 root netem delay 300ms 100ms

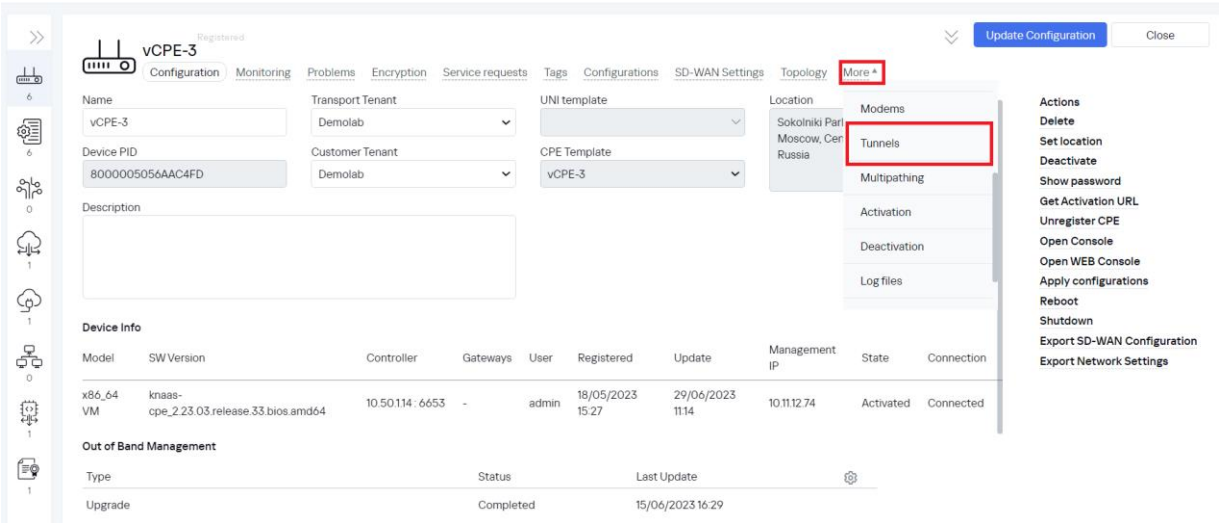
и снять задержку с первого сетевого интерфейса:

[root@isp]# tc qdisc del dev ens193 root

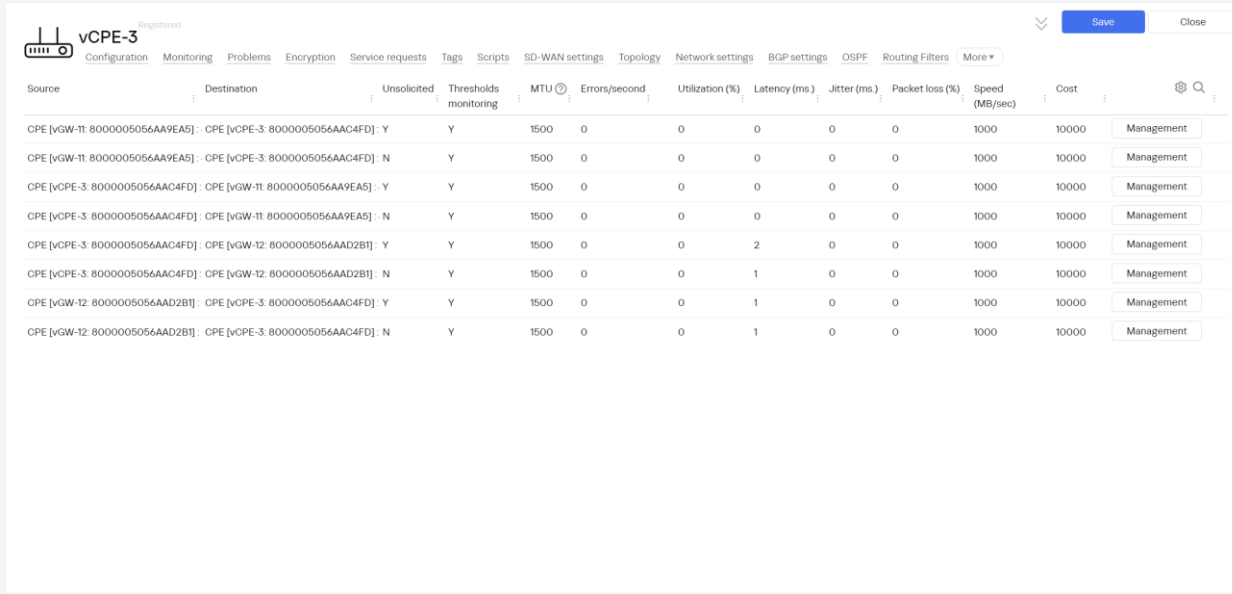
3.5.4. Перейти в меню CPE и выбрать vCPE-3.



3.5.5. Перейти на вкладку Tunnels.



Отобразится список построенных туннелей с CPE.



3.5.6. Включение мониторинга задержек на туннелях, установленных vCPE-3.

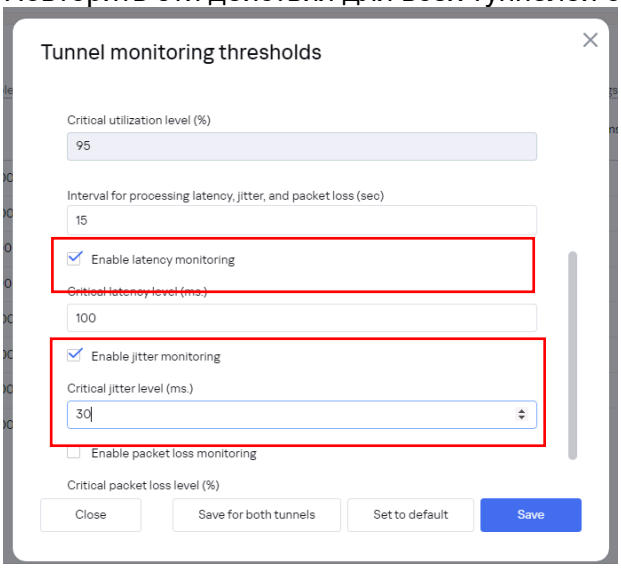
Нажать Management > Set monitoring thresholds.

Включить:

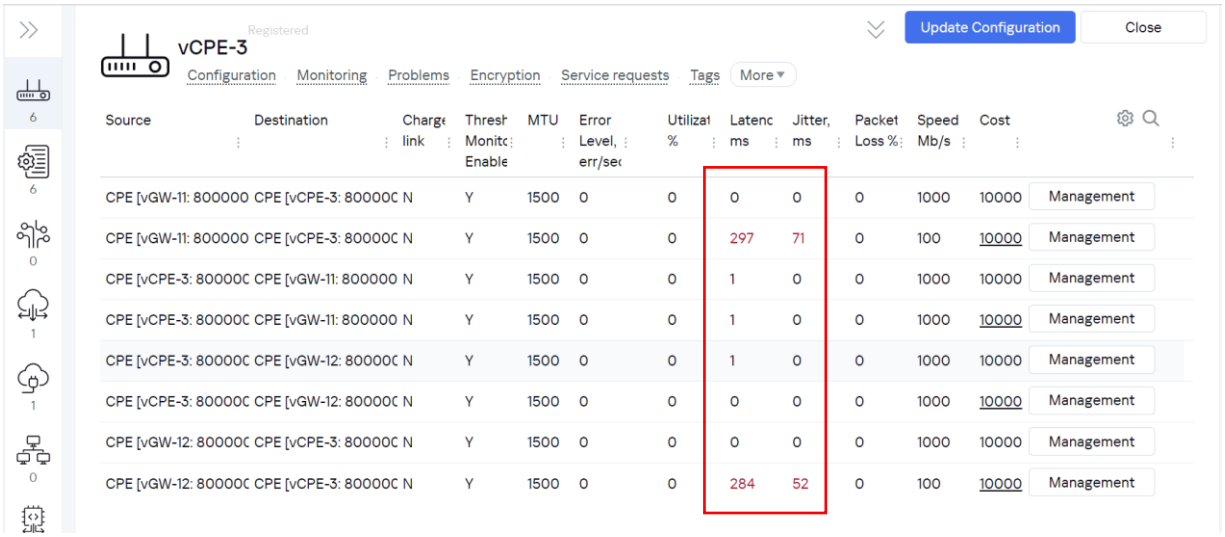
- Enable tunnel thresholds monitoring
- Enable latency monitoring > Critical latency level – 100 msec.
- Enable jittter monitoring > Critical jitter level – 30 msec.

Нажать Save for both links – сохранение параметров мониторинга туннелей в оба направления.

Данные настройки включат мониторинг задержки и джиттера для туннелей и зададут пороговые значения в 30мс и 100мс соответственно.
Повторить эти действия для всех туннелей с данным CPE устройством.

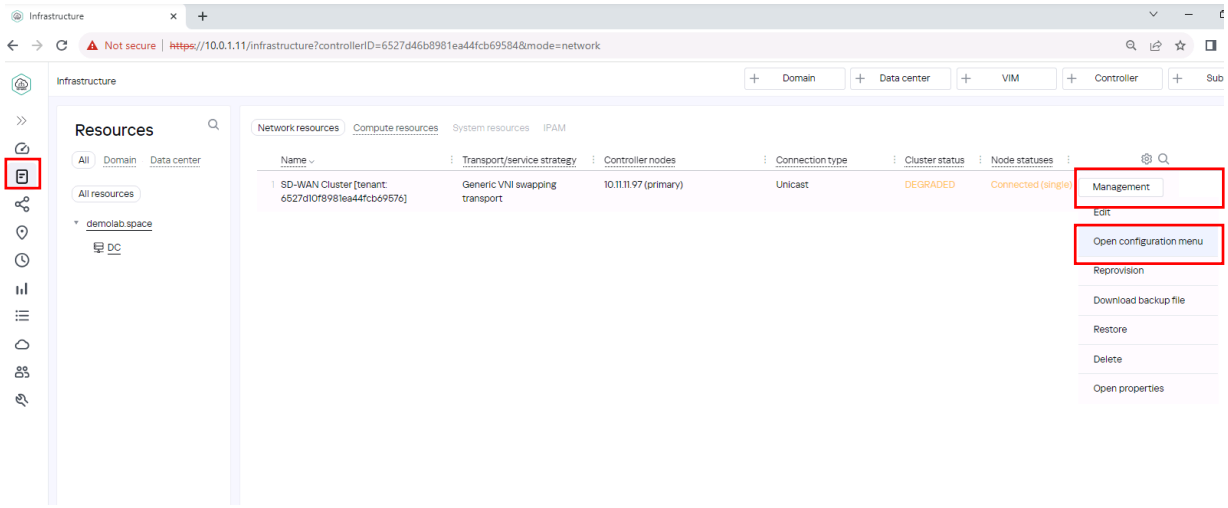


3.5.7. После применения настроек в п. 3.5.6 отобразится статистика задержек и джиттера на туннелях. Значения измеренных параметров, не удовлетворяющих порогам, заданных в 3.5.6 будут выделены красным цветом.



3.5.8. Создание ограничений Constraints.

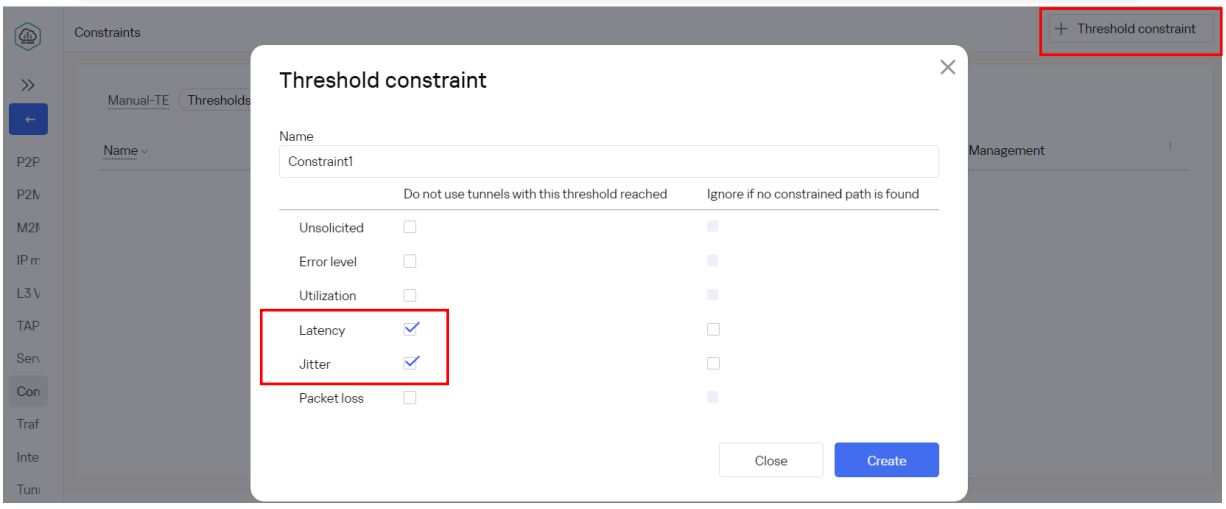
Для перенаправления трафика необходимо создать ограничения (Constraints).
Перейти в меню Infrastructure > SD-WAN контроллер > Management >open configuration menu.



Перейти в меню Constraints, затем открыть вкладку Thresholds и нажать на кнопку *+Threshold Constraint*.

Задать название Constraints в поле name и включить ограничение для задержки (latency) и джиттера (jitter).

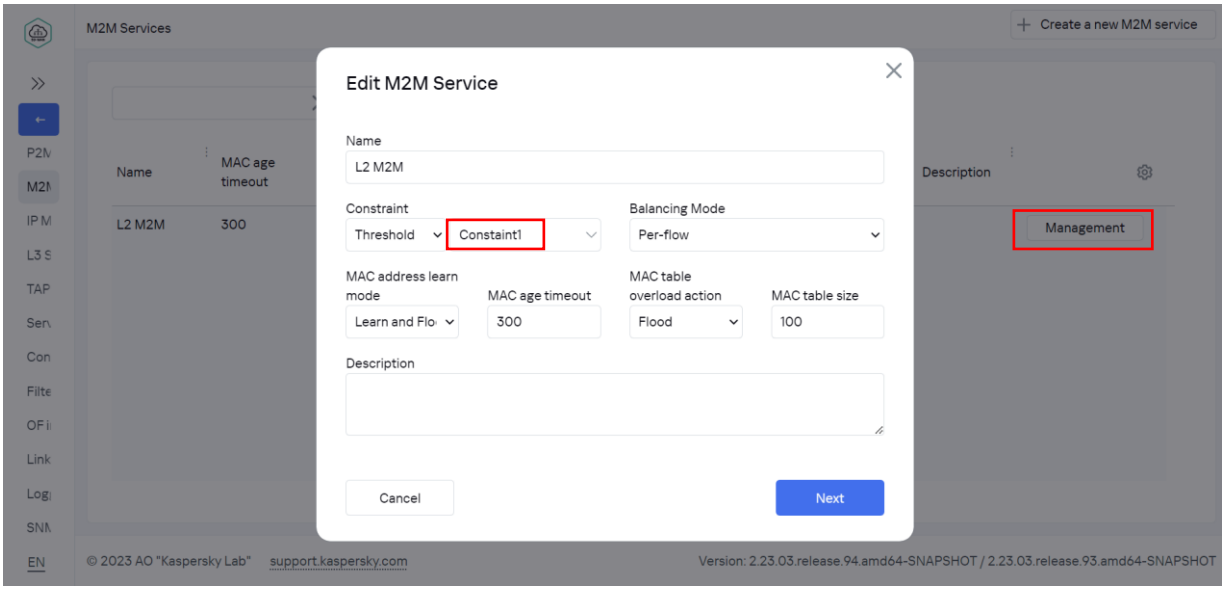
Данное ограничение исключит из путей прохождения трафика туннели, не отвечающие настроенным в п. 3.5.6 пороговым значениям.



3.5.9. Применение ограничений(constraints) к транспортному сервису.

Перейти на вкладку M2M Services. Открыть сервис для редактирования: Management > Edit.

Выбрать созданное в п. 3.5.8 ограничение в секции Constraint. Нажать Next, Next, Save.



3.5.10. После применения Constraints контроллер SD-WAN уберет трафик с туннелей, не удовлетворяющий ограничению (Constraint), примененному в 3.5.9.

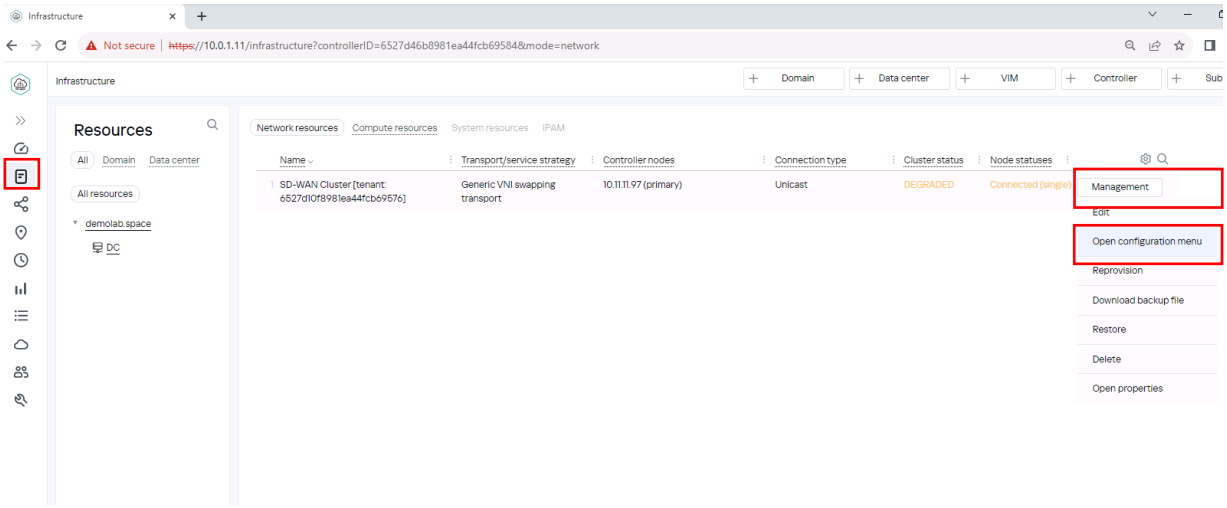
В статистике iperf на wst4 наглядно видно, что джиттер пропал, потому что SD-WAN контроллер исключил туннели, проходящие через первый WAN интерфейс vCPE-3, для которого была применена эмуляция latency и jitter.

```
[ ID] Interval           Transfer     Bandwidth         Jitter    Lost/Total Datagrams
[  5]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  0.102 ms  0/82 (0%)
[  5]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  0.102 ms  0/91 (0%)
[  5]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  0.109 ms  0/90 (0%)
[  5]   3.00-4.00   sec   129 KBytes  1.05 Mbits/sec  0.128 ms  0/91 (0%)
[  5]   4.00-5.00   sec   127 KBytes  1.04 Mbits/sec  0.133 ms  0/90 (0%)
[  5]   5.00-6.00   sec   129 KBytes  1.05 Mbits/sec  0.110 ms  0/91 (0%)
[  5]   6.00-7.00   sec   127 KBytes  1.04 Mbits/sec  0.107 ms  0/90 (0%)
[  5]   7.00-8.00   sec   129 KBytes  1.05 Mbits/sec  0.142 ms  0/91 (0%)
[  5]   8.00-9.00   sec   127 KBytes  1.04 Mbits/sec  0.120 ms  0/90 (0%)
[  5]   9.00-10.00  sec   129 KBytes  1.05 Mbits/sec  0.096 ms  0/91 (0%)
[  5]  10.00-11.00  sec   127 KBytes  1.04 Mbits/sec  0.133 ms  0/90 (0%)
[  5]  11.00-12.00  sec   129 KBytes  1.05 Mbits/sec  0.134 ms  0/91 (0%)
[  5]  12.00-13.00  sec   127 KBytes  1.04 Mbits/sec  0.143 ms  0/90 (0%)
[  5]  13.00-14.00  sec   129 KBytes  1.05 Mbits/sec  0.169 ms  0/91 (0%)
[  5]  14.00-15.00  sec   127 KBytes  1.04 Mbits/sec  0.167 ms  0/90 (0%)
[  5]  15.00-16.00  sec   129 KBytes  1.05 Mbits/sec  0.114 ms  0/91 (0%)
[  5]  16.00-17.00  sec   127 KBytes  1.04 Mbits/sec  0.117 ms  0/90 (0%)
[  5]  17.00-18.00  sec   129 KBytes  1.05 Mbits/sec  0.092 ms  0/91 (0%)
[  5]  18.00-19.00  sec   127 KBytes  1.04 Mbits/sec  0.095 ms  0/90 (0%)
[  5]  19.00-20.00  sec   129 KBytes  1.05 Mbits/sec  0.116 ms  0/91 (0%)
[  5]  20.00-21.00  sec   127 KBytes  1.04 Mbits/sec  0.128 ms  0/90 (0%)
[  5]  21.00-22.00  sec   129 KBytes  1.05 Mbits/sec  0.147 ms  0/91 (0%)
[  5]  22.00-23.00  sec   127 KBytes  1.04 Mbits/sec  0.107 ms  0/90 (0%)
[  5]  23.00-24.00  sec   129 KBytes  1.05 Mbits/sec  0.133 ms  0/91 (0%)
[  5]  24.00-25.00  sec   127 KBytes  1.04 Mbits/sec  0.118 ms  0/90 (0%)
[  5]  25.00-26.00  sec   129 KBytes  1.05 Mbits/sec  0.105 ms  0/91 (0%)
[  5]  26.00-27.00  sec   129 KBytes  1.05 Mbits/sec  0.091 ms  0/91 (0%)
[  5]  27.00-28.00  sec   127 KBytes  1.04 Mbits/sec  0.171 ms  0/90 (0%)
```

3.5.11. Возврат настроек после завершения теста.

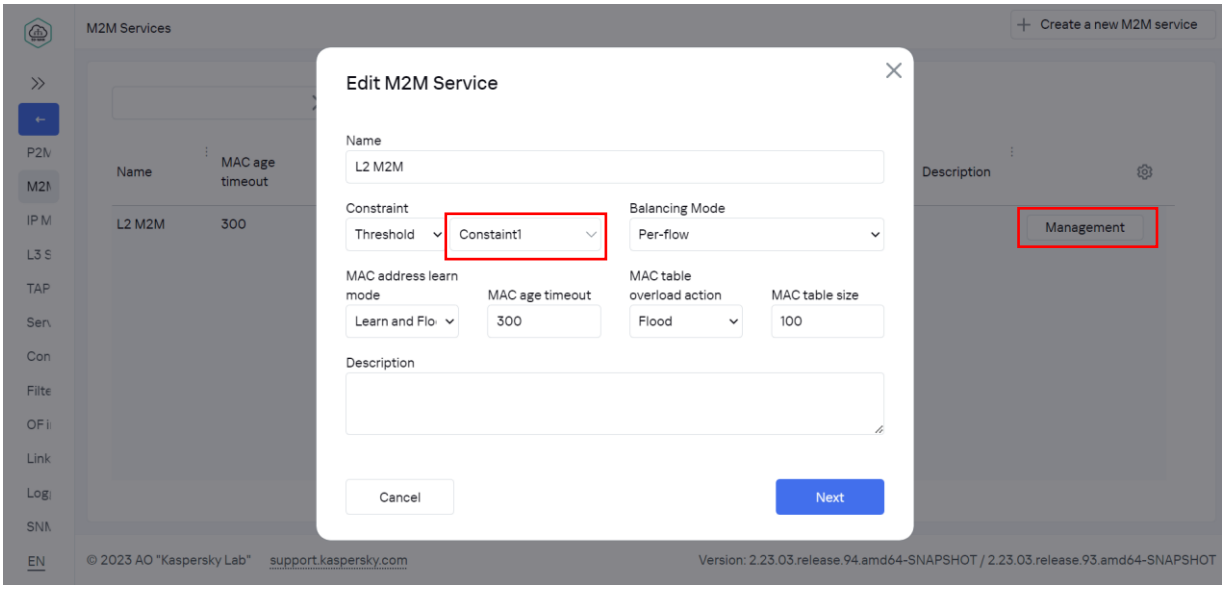
Снять ограничение с транспортного сервиса, примененного в п. 3.5.9.

Перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



Перейти на вкладку M2M Services. Открыть сервис L2 M2M для редактирования: Management > Edit.

Убрать ограничение в секции Constraint. Нажать Next, Next, Save.



Выключить эмуляцию задержек и джиттера на хосте isp.

[root@isp]# tc qdisc del dev ens161 root

[root@isp]# tc qdisc del dev ens193 root

3.6. **Приоритезация трафика с использованием ACL.**

Решение SD-WAN позволяет создавать классификаторы трафика на основе полей заголовков IP/TCP/UDP и направлять трафик в определенные транспортные сервисы. Например, возможно создать приоритетный сервис для чувствительного к задержке трафика с ограничениями, чтобы трафик не проходил через туннели с задержкой, не удовлетворяющей заданному ограничению.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Направление трафика приложения в транспортный сервис:

https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/246544.htm

В данном сценарии создается классификатор трафика на основе UDP порта для перенаправления тестового трафика в приоритетный сервис.

Тестовый трафик будет генерироваться между рабочими станциями wst3 и wst4 с использованием iperf на порту UDP 5555. Будет создан L3 ACL для классификации тестового трафика и “ACL интерфейс” для перенаправления трафика отдельный сервис.

В данном сценарии, туннели, проходящие через интерфейс sdwan0 (eth0) vCPE-3 будут отмечены как “ Unsolicited ” («нежелательный» для использования) и создан отдельный транспортный сервис. Для отдельного сервиса будут заданы ограничения (Constraints), которые исключат туннели, отмеченные как Unsolicited из пути прохождения трафика. Для проверки переключения трафика будет использоваться tcpdump на vCPE-3.

3.6.1. Генерация тестового трафика.

Запустить сервер iperf на хосте wst4 портом 5555:

[root@wst4]# iperf3 -s -p 5555

```
^Ciperf3: interrupt - the server has terminated
[ivpanin@wst4 ~]$ iperf3 -s -p 5555
-----------------------------------------------------------
Server listening on 5555
-----------------------------------------------------------
```

Запустить клиент iperf на хосте wst3 с портом 5555:

[root@wst3]# iperf3 -u -t 6000 -c 10.20.4.11 -p 5555

```
[ivpanin@wst3 ~]$ iperf3 -u -t 6000 -c 10.20.4.11 -p 5555
Connecting to host 10.20.4.11, port 5555
[  4] local 10.20.3.11 port 41924 connected to 10.20.4.11 port 5555
[ ID] Interval           Transfer     Bandwidth       Total Datagrams
[  4]   0.00-1.00   sec   116 KBytes   950 Kbits/sec  82
[  4]   1.00-2.00   sec   129 KBytes  1.05 Mbits/sec  91
[  4]   2.00-3.00   sec   127 KBytes  1.04 Mbits/sec  90
```

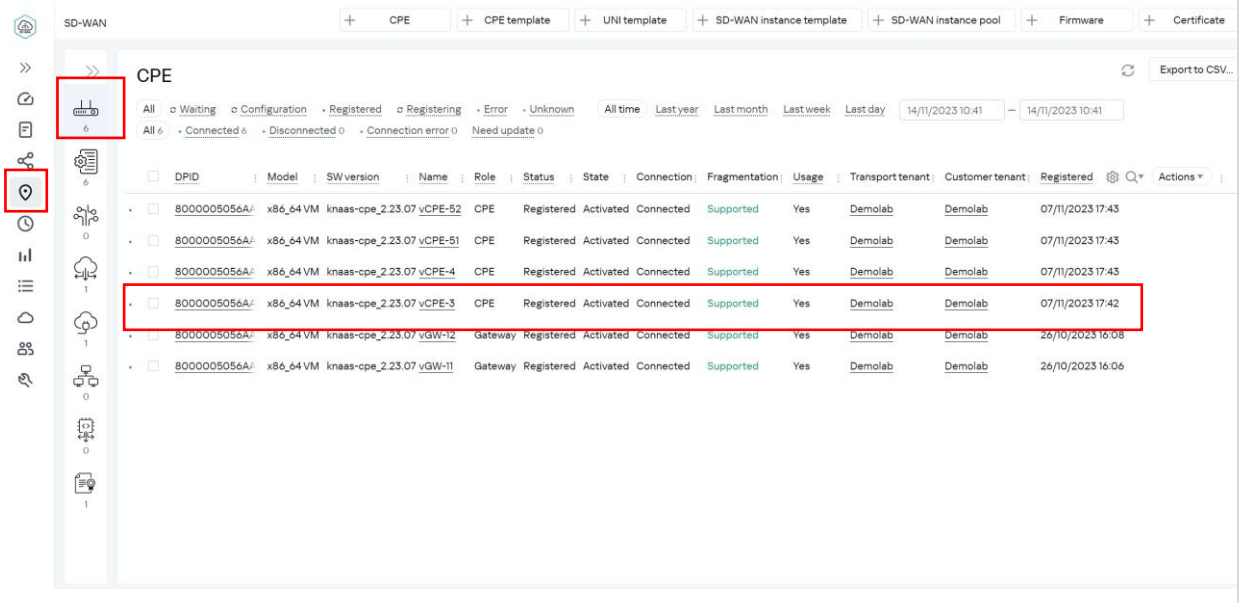
3.6.2. Подключиться к vCPE-3 по SSH и найти через какой интерфейс проходит трафик: *genev_sys_4800* или *genev_sys_4801*:

 # tcpdump -i genev_sys_4800
genev_sys – туннельные интерфейсы CPE. Номер порта указывает на номер WAN интерфейса CPE устройства. Номера назначаются по порядку, начиная с порта 4800, по одному на каждый WAN интерфейс. Порт 4800 означает WAN интерфейс sdwan0 (eth0), порт 4801 означает WAN интерфейс sdwan1 (eth1).

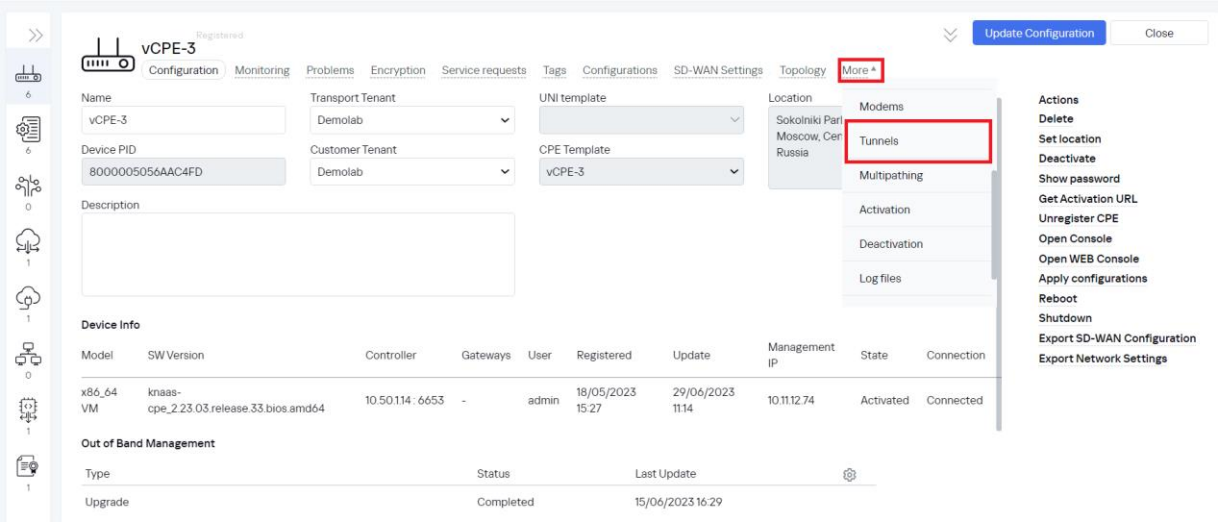
В данном примере трафик проходит через интерфейс *genev_sys_4800*.

```
root@8000005056AAC4FD:~# tcpdump -i genev_sys_4800
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on genev_sys_4800, link-type EN10MB (Ethernet), capture size 262144 bytes
08:31:03.145408 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145467 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145491 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145518 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145539 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145564 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145800 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145825 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.145849 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245488 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245582 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245608 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245629 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245659 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245928 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245953 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245976 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
08:31:03.245996 IP 10.20.3.11.41924 > 10.20.4.11.5555: UDP, length 1448
```

3.6.3. Перейти в меню CPE и выбрать vCPE-3.



3.6.4. Перейти на вкладку Tunnels.



3.6.5. Задание параметра “Unsolicited” для туннелей.

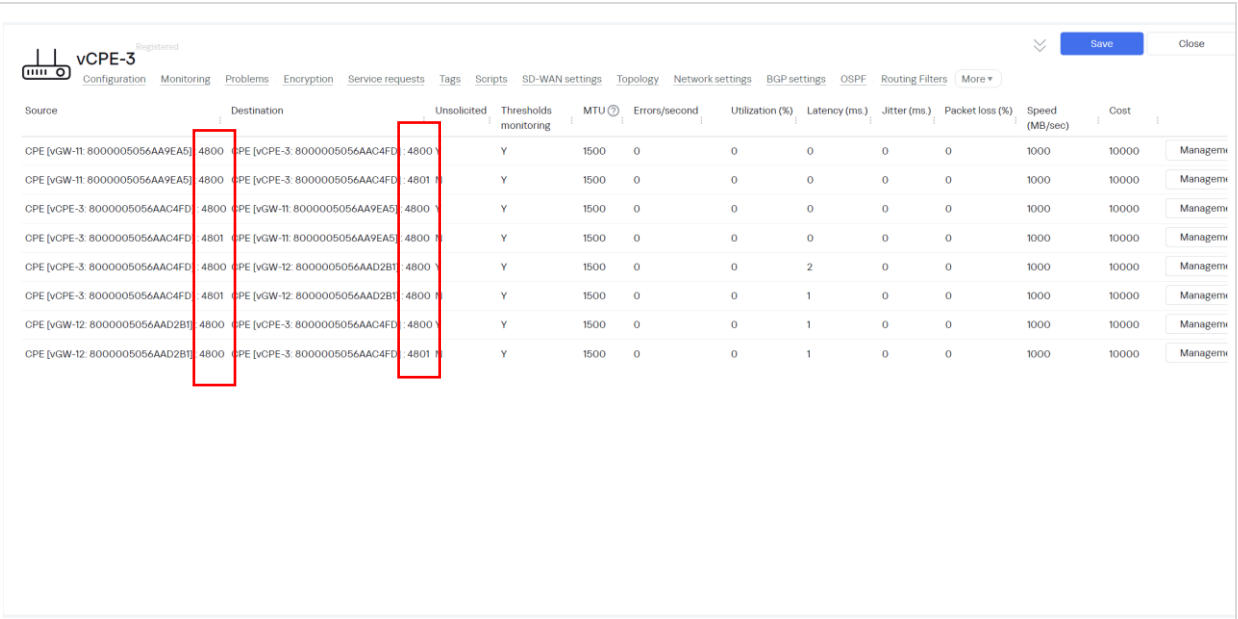
Найти все туннели, через которые проходит трафик: порты источника и назначения туннелей (4800 или 4801) должны совпадать с номером интерфейса согласно проверке в пункте 3.6.2. В результате проверки в данном примере трафик проходит через туннель *genev_sys_4800*.
Туннели, через которые проходит трафик в примере:

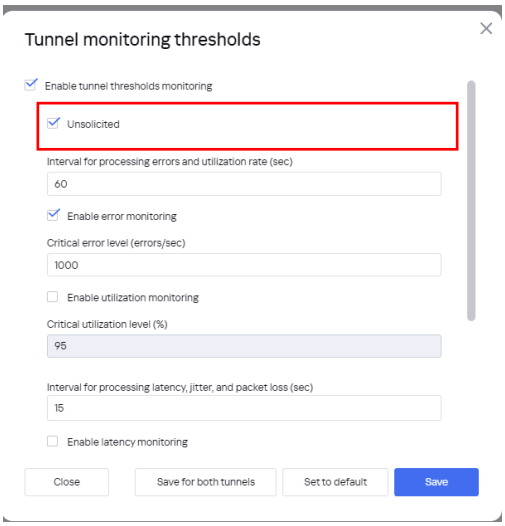
- vCPE-3:4800 <--> vGW-11:4800
- vCPE-3:4800 <--> vGW-12:4800
- vGW-11:4800 <--> vCPE-3:4800
- vGW-12:4800 <--> vCPE-3:4800

Поочередно для каждого найденного туннеля с портом 4800 для vCPE-3 нажать Management > Set tunnel monitoring threshold.

Отметить туннель как “ Unsolicited” – означает «нежелательный» для использования.

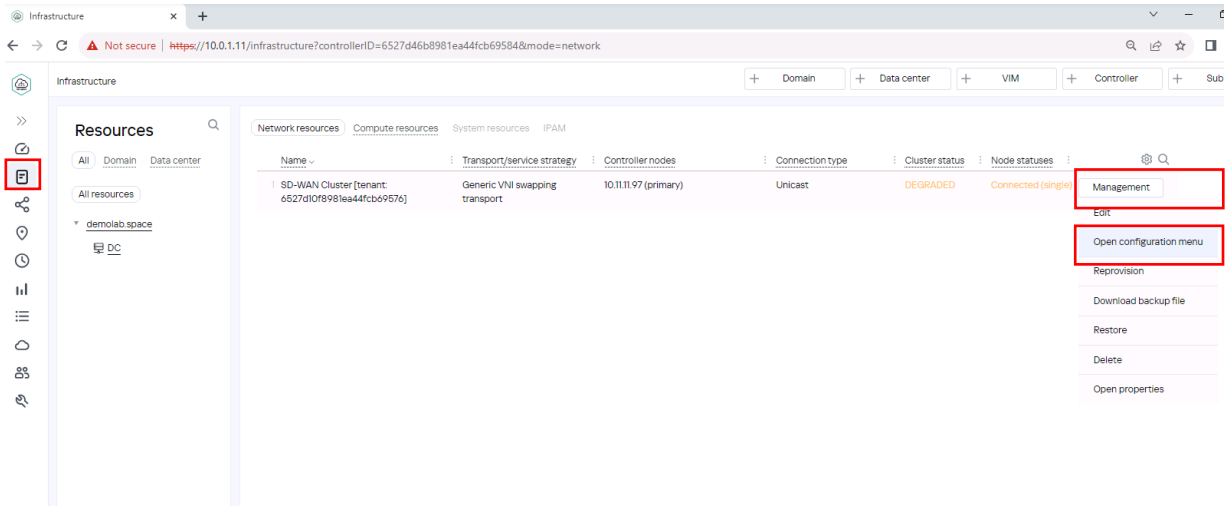
Нажать Save for both tunnels – сохранение параметров мониторинга туннелей в оба направления.





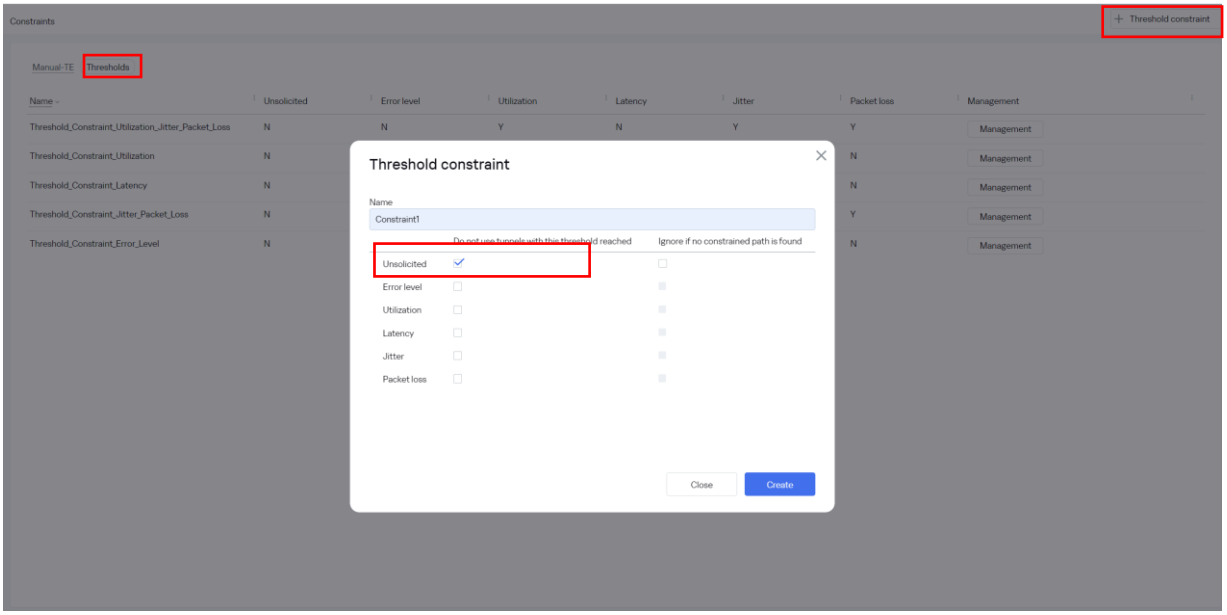
3.6.6. Создание Constraints.

Для перенаправления трафика необходимо создать ограничения. Перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



Перейти в меню Constraints/ затем отрыть вкладку Thresholds и нажать на кнопку *+Threshold Constraint*.

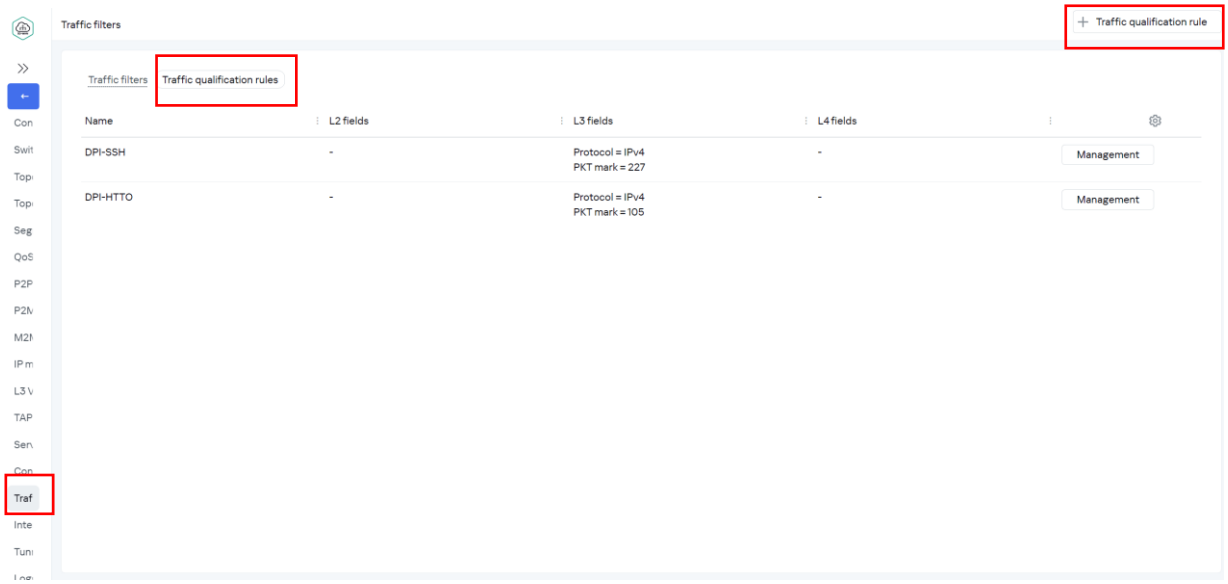
Задать название Constraints в поле name и включить ограничение Unsolicited Данное ограничение исключит из транспортного сервиса туннели, отмеченные как Unsolicited. Нажать Save.



3.6.7. Создание правил фильтрации.

Для направления трафика в отдельный сервис нужно создать список доступа ACL, чтобы поймать тестовый трафик UDP с портом 5555.

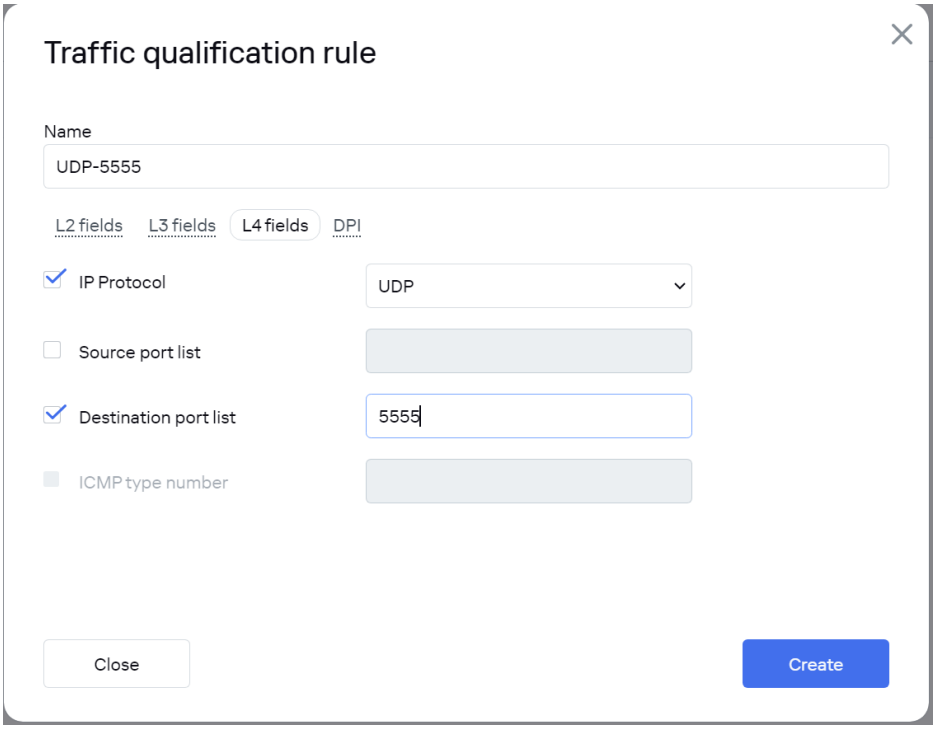
Перейти в меню вкладку Traffic Filters. Затем зайти во вкладку Traffic qualification rules и нажать +Traffic qualification rules.



Задать имя Rule:

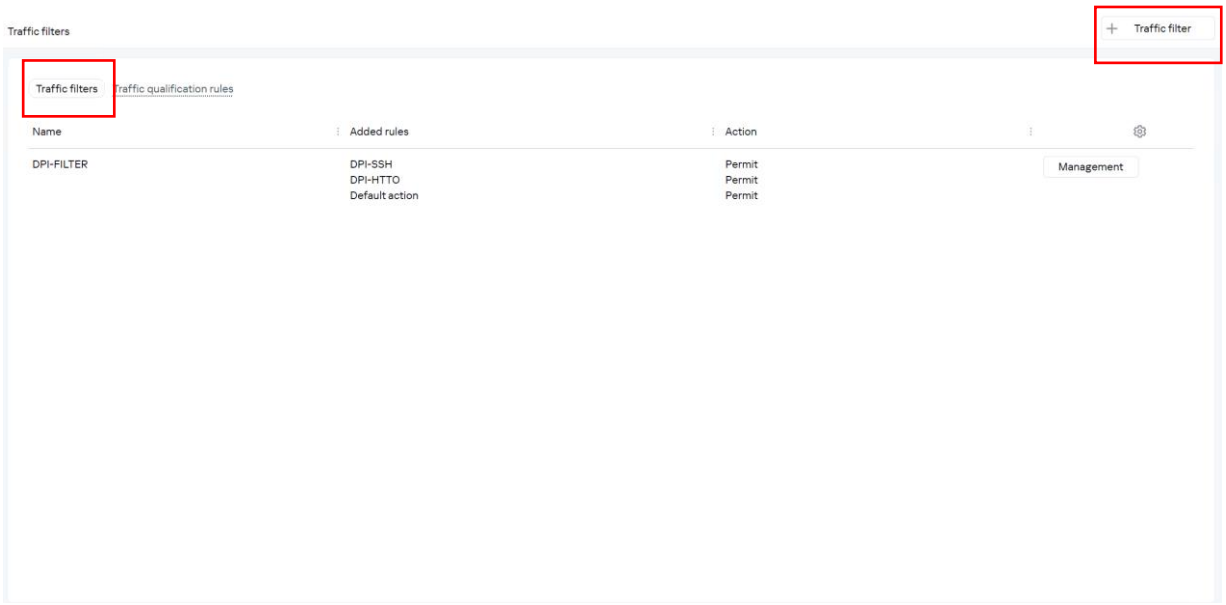
- Отметить во вкладке L3 Fields: Protocol – IPv4.
- Отметить во вкладке L4 Fields:
- IP protocol – UDP.
- Destination port -5555.

Нажать Create.

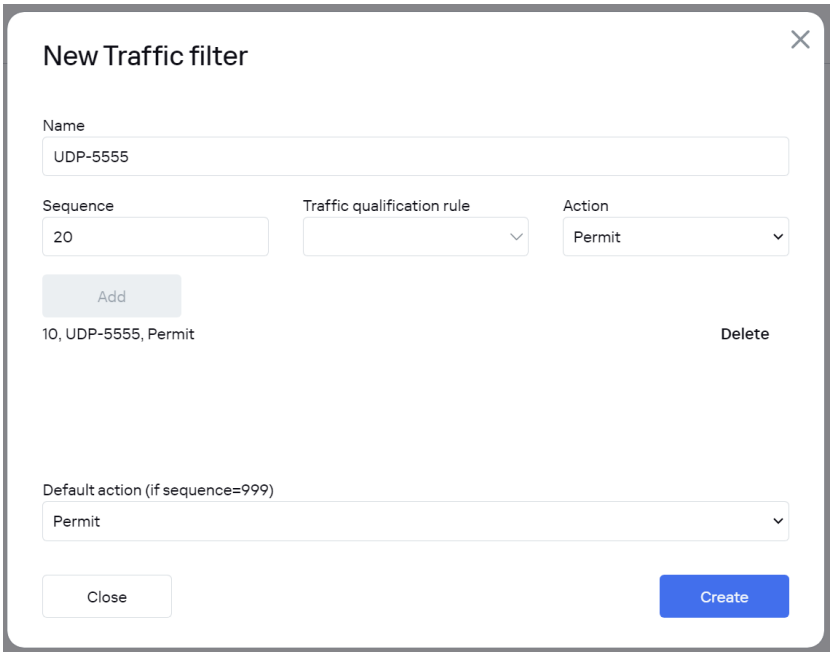


3.6.8. Создание фильтра.

Перейти на вкладку Traffic filters, нажать *+Traffic filter.*



- Задать имя фильтра.
- Добавить правила, созданные в п. 3.6.7. Выбрать в селекторе Rule созданное 3.6.7. правило, задать Action – Permit. Нажать Add.
- нажать Create.



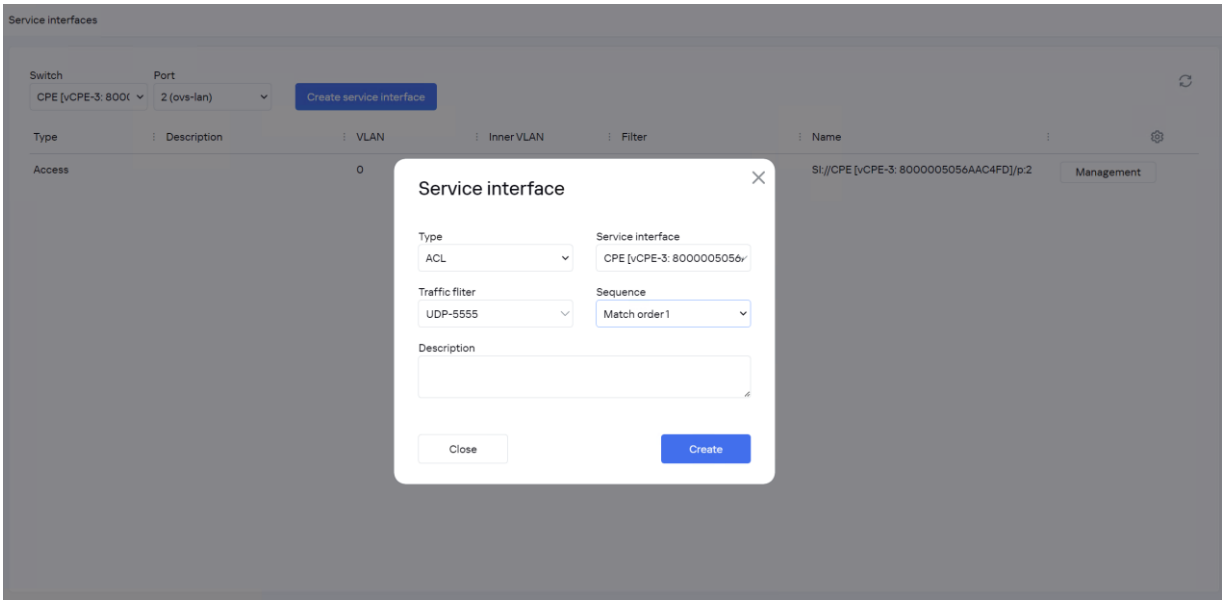
3.6.9. Создание ACL Service interfaces.

Трафик попадает в транспортный сервис через сервисные интерфейсы. Необходимо создать специальный ACL интерфейс (ACL Service Interface – ACL SI). Перейти на вкладку Service Interfaces, затем выбрать Switch vCPE-3 и Port 2 (ovs-lan).

Нажать Create service interface. Выбрать:

- Type: ACL.
- Service interface: vCPE-3 (port 2).
  Traffic Filter с UDP 5555, созданный в пункте 3.6.8.
- Sequence “*Match order 1”* (данный ACL SI будет с первым обрабатывать трафик).

Нажать Create.



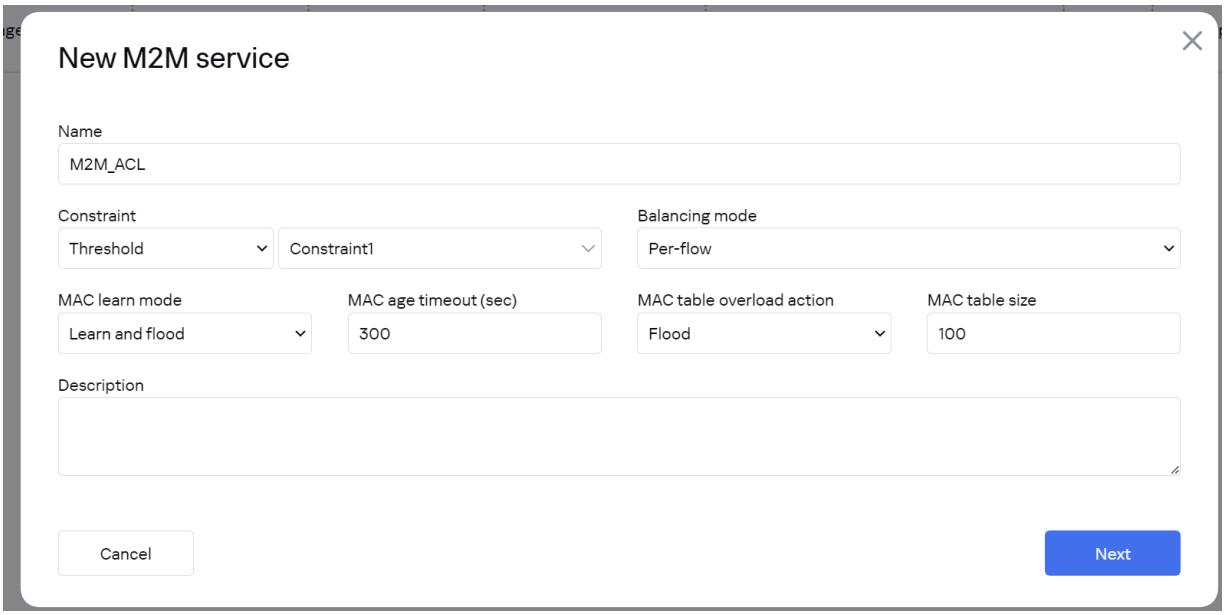
3.6.10. Для создания сервиса требуется создать сервисные интерфейсы для каждой CPE.

Повторить п.3.6.9 для vCPE-4.

3.6.11. Создание отдельного транспортного сервиса для приоритетного трафика.

Перейти в M2M Service, нажать +Create a new M2M service.

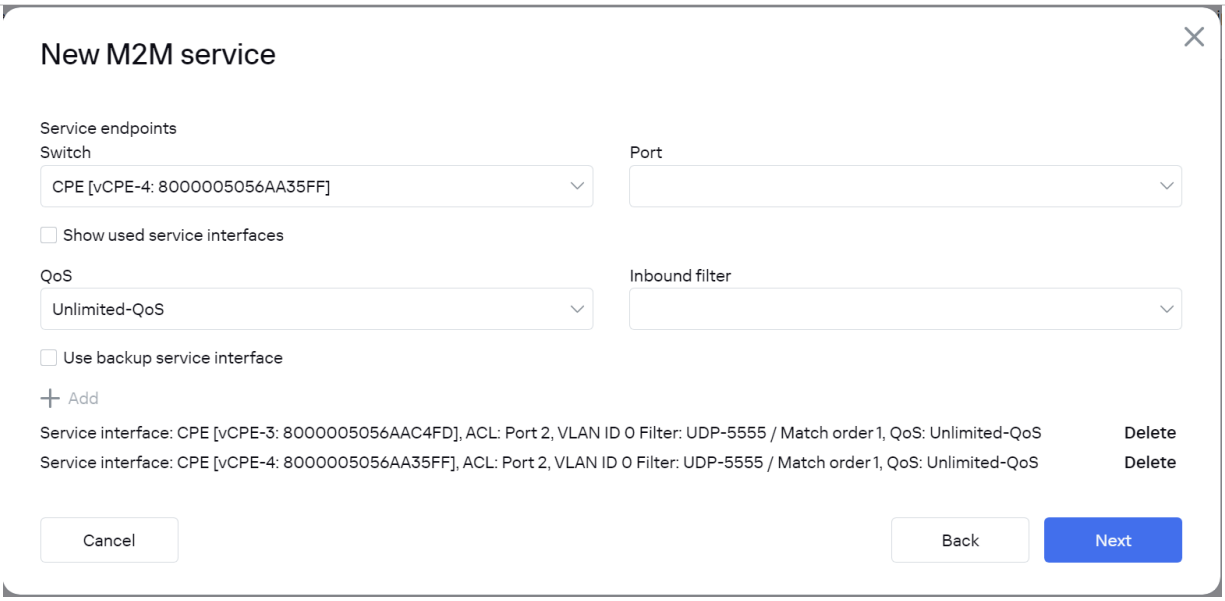
Задать название, выбрать созданный ранее Threshold (в пункте 3.6.6), нажать Next.



Добавить 2 сервисных интерфейса (для vCPE-3 и vCPE-4) для направления трафика в сервис:

- выбрать Switch (vCPE3 и vCPE4)
- Port – созданные в п.3.6.9 и 3.6.10 ACL SI
- Qos – Unlimited-QoS
- Нажать на +Add

Нажать Next и Create.



3.6.12. Подключиться к vCPE-3 и проверить, что трафик переключился на другой WAN интерфейс (в зависимости от настроек, сделанных ранее):
В пункте 3.6.2 проверялось, что трафик идёт через интерфейс genev_sys_4800 (sdwan0). После настройки отдельного транспортного сервиса в результате работы ограничений и фильтра трафик перешел на интерфейс genev_sys_4801 (sdwan1).

Проверить с помощью tcpdump наличие трафика на интерфейсе geneve_sys_4801.
 # tcpdump -i genev_sys_4801

На скриншоте видно, что трафик переключился с интерфейса genev_sys_4800 (sdwan0) на genev_sys_4801 (sdwan1).

```
root@8000005056AAC4FD:~# tcpdump -i genev_sys_4801
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on genev_sys_4801, link-type EN10MB (Ethernet), capture size 262144 bytes
08:54:33.012984 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013061 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013631 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013665 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013689 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013710 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013730 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013750 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.013770 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.112997 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113067 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113529 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113566 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113591 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113617 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113638 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113668 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.113691 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214098 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214651 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214700 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214724 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214751 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214776 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214803 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
08:54:33.214832 IP 10.20.3.11.53775 > 10.20.4.11.5555: UDP, length 1448
```

3.6.13. Возврат настроек после завершения теста.

Удалить сервис, созданный в п. 3.6.11 (при удалении отметить Delete associated service interfaces).

Убрать параметр “Unsolicited” с туннелей, добавленный в п. 3.6.5.

Остановить iperf, запущенный в п. 3.6.1.

3.7. **Приоритезация трафика с использованием DPI.**

Решение SD-WAN позволяет создавать классификаторы трафика с помощью DPI и перенаправлять трафик для определенных приложений.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Направление трафика приложения в транспортный сервис:

https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/246544.htm

В данном сценарии создается классификатор для SSH и HTTP трафика для перенаправления в приоритетный сервис. Тестовый трафик будет генерироваться между рабочими станциями wst3 до wst4 с использованием ssh, nc и curl. Будет создан DPI правило для классификации тестового трафика и ACL интерфейс для перенаправления трафика в отдельный сервис. Туннели, проходящие через интерфейс sdwan0 (eth0) vCPE-3 будут отмечены как Unsolicited, также будет создан отдельный транспортный сервис, для которого будут заданы ограничения (Constraints), которые исключат Unsolicited туннели из пути прохождения трафика. Для проверки переключения трафика будет использоваться tcpdump на vCPE-3.

3.7.1. Для теста необходимо запустить сессию SSH на хосте wst3 до wst4:

[root@wst3]# ssh root@10.20.4.11

```
[ivpanin@wst3 ~]$ ssh root@10.20.4.11
root@10.20.4.11's password:
Last login: Mon Jun 26 16:32:13 2023 from 10.20.3.11
[root@wst4 ~]#
```

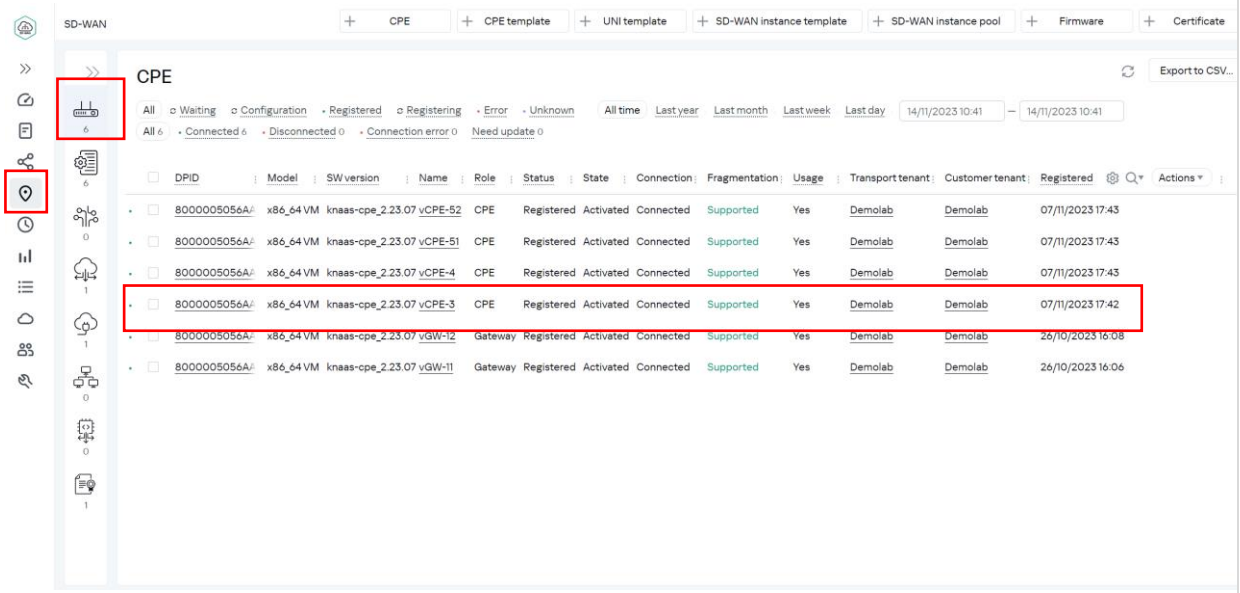
3.7.2. Подключиться к vCPE-3 и проверить, что трафик идет через один из интерфейсов *genev_sys_4800 или genev_sys_4801*:

 # tcpdump -i genev_sys_4800
genev_sys – туннельные интерфейсы CPE. Номер порта указывает на номер WAN интерфейса CPE устройства. Номера назначаются по порядку, начиная с порта 4800, по одному на каждый WAN интерфейс. Порт 4800 означает WAN интерфейс sdwan0 (eth0), порт 4801 означает WAN интерфейс sdwan1 (eth1).

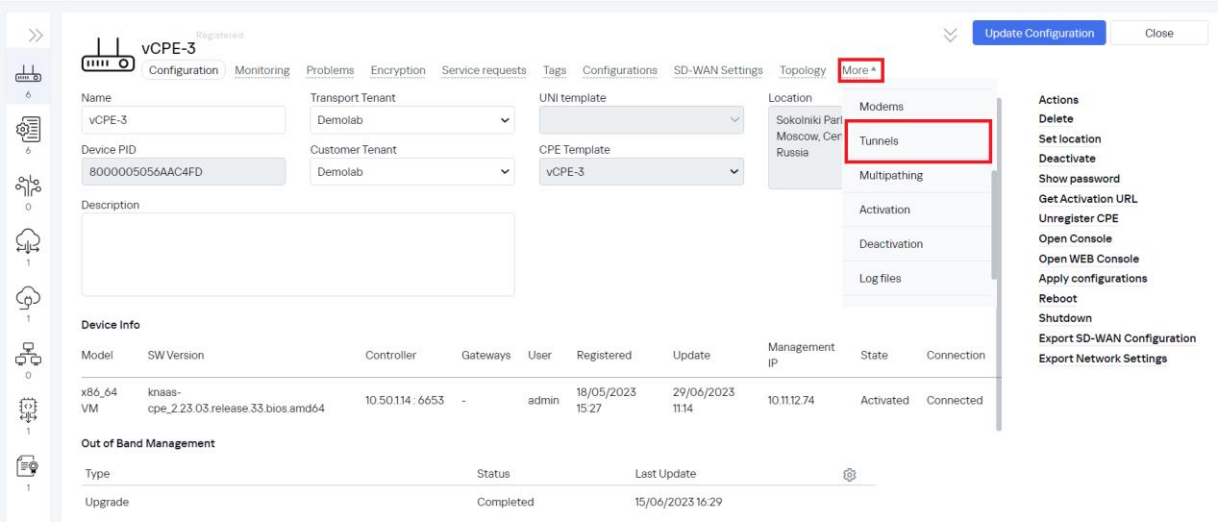
Запомнить на каком из интерфейсов сейчас идет тестовой трафик, для SSH сессии он может быть ассиметричным (в одну сторону через 4800, а в другую через 4801). В примере SSH идет через *genev_sys_4800 (sdwan0).*

```
root@8000005056AAC4FD:~# tcpdump -i genev_sys_4800 | grep 4.11
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on genev_sys_4800, link-type EN10MB (Ethernet), capture size 262144 bytes
13:39:15.283859 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29978, length 64
13:39:15.284559 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29978, length 64
13:39:16.285837 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29979, length 64
13:39:16.286389 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29979, length 64
13:39:17.287721 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29980, length 64
13:39:17.288264 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29980, length 64
13:39:18.289822 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29981, length 64
13:39:18.290502 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29981, length 64
13:39:19.291081 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29982, length 64
13:39:19.291723 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29982, length 64
13:39:20.292795 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29983, length 64
13:39:20.293434 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29983, length 64
13:39:20.680213 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [P.], seq 619880975:619881011, ack 1077224629, win 291, options [nop,nop,TS val 235266084 ecr 235
218972], length 36
13:39:20.684756 IP 10.20.4.11.ssh > 10.20.3.11.56790: Flags [P.], seq 1:37, ack 36, win 295, options [nop,nop,TS val 235309077 ecr 235266084], length 36
13:39:20.685333 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [.], ack 37, win 291, options [nop,nop,TS val 235266090 ecr 235309077], length 0
13:39:20.811386 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [P.], seq 36:72, ack 37, win 291, options [nop,nop,TS val 235266216 ecr 235309077], length 36
13:39:20.814117 IP 10.20.4.11.ssh > 10.20.3.11.56790: Flags [P.], seq 37:73, ack 72, win 295, options [nop,nop,TS val 235309207 ecr 235266216], length 36
13:39:20.814651 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [.], ack 73, win 291, options [nop,nop,TS val 235266219 ecr 235309207], length 0
13:39:20.884017 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [P.], seq 72:108, ack 73, win 291, options [nop,nop,TS val 235266288 ecr 235309207], length 36
13:39:20.886770 IP 10.20.4.11.ssh > 10.20.3.11.56790: Flags [P.], seq 73:109, ack 108, win 295, options [nop,nop,TS val 235309280 ecr 235266288], length 36
13:39:20.887343 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [.], ack 109, win 291, options [nop,nop,TS val 235266292 ecr 235309280], length 0
13:39:21.085620 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [P.], seq 108:144, ack 109, win 291, options [nop,nop,TS val 235266490 ecr 235309280], length 36
13:39:21.088901 IP 10.20.4.11.ssh > 10.20.3.11.56790: Flags [P.], seq 109:185, ack 144, win 295, options [nop,nop,TS val 235309482 ecr 235266490], length 76
13:39:21.089380 IP 10.20.3.11.56790 > 10.20.4.11.ssh: Flags [.], ack 185, win 291, options [nop,nop,TS val 235266494 ecr 235309482], length 0
13:39:21.294580 IP 10.20.4.11 > 10.20.3.11: ICMP echo request, id 5153, seq 29984, length 64
13:39:21.295229 IP 10.20.3.11 > 10.20.4.11: ICMP echo reply, id 5153, seq 29984, length 64
13:39:21.504411 IP 10.0.1.11.36582 > 10.11.12.74.zabbix-agent: Flags [.], ack 2143178383, win 502, options [nop,nop,TS val 4169110840 ecr 3932987910], length
0
```

3.7.3. Перейти в меню CPE и выбрать vCPE-3.



3.7.4. Перейти на вкладку Tunnels.



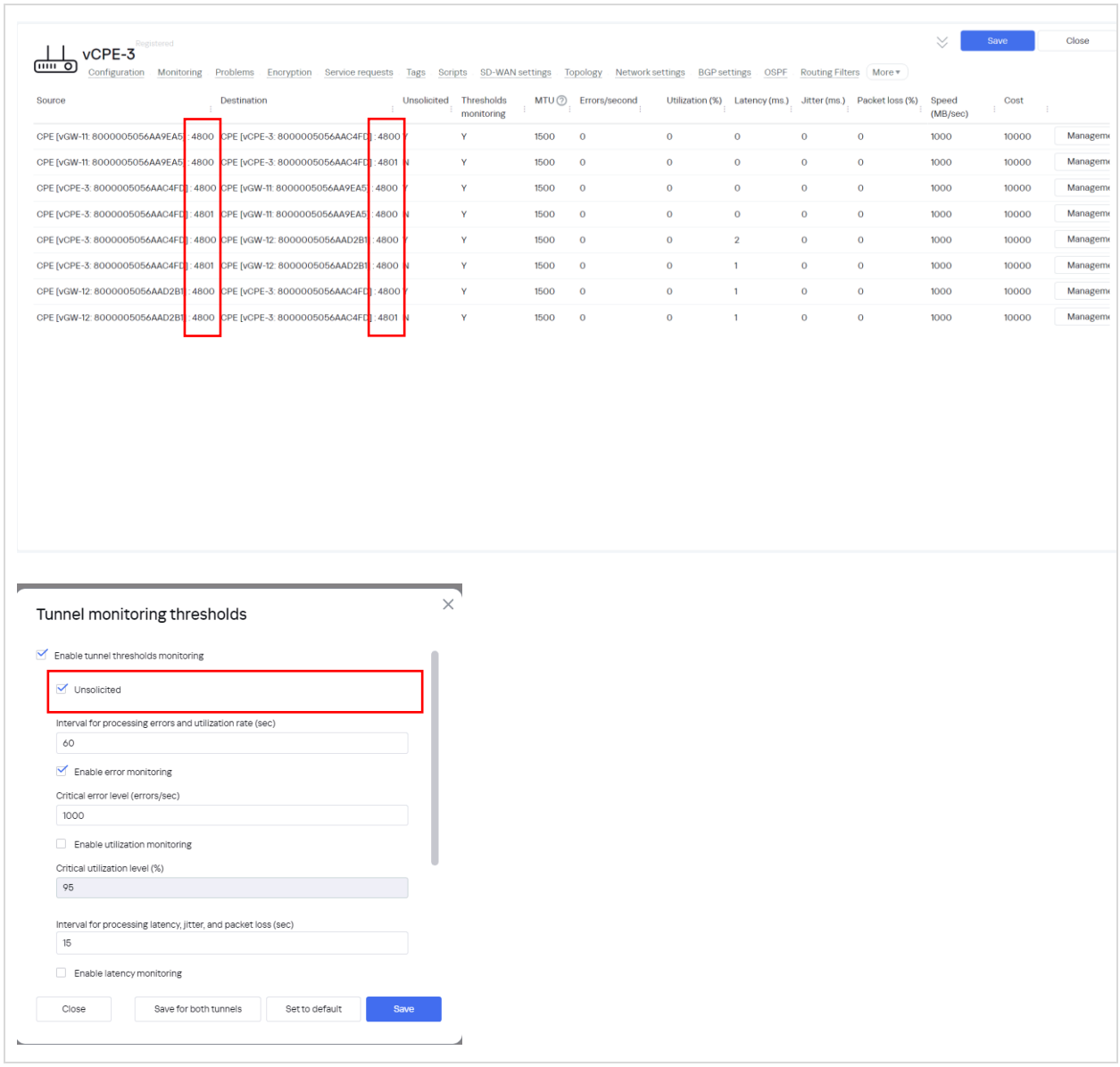
3.7.5. Задание параметра “ Unsolicited ” для туннелей.
Найти все туннели, через которые проходит трафик: порты источника и назначения туннелей (4800 или 4801) должны совпадать с номером интерфейса согласно проверке в пункте 3.7.2. В результате проверки в данном примере трафик проходит через туннель *genev_sys_4800*.
Туннели, через которые проходит трафик в примере:

- vCPE-3:4800 <--> vGW-11:4800
- vCPE-3:4800 <--> vGW-12:4800
- vGW-11:4800 <--> vCPE-3:4800
- vGW-12:4800 <--> vCPE-3:4800

Поочередно для каждого найденного туннеля с портом 4800 для vCPE-3 нажать Management > Set tunnel monitoring threshold.

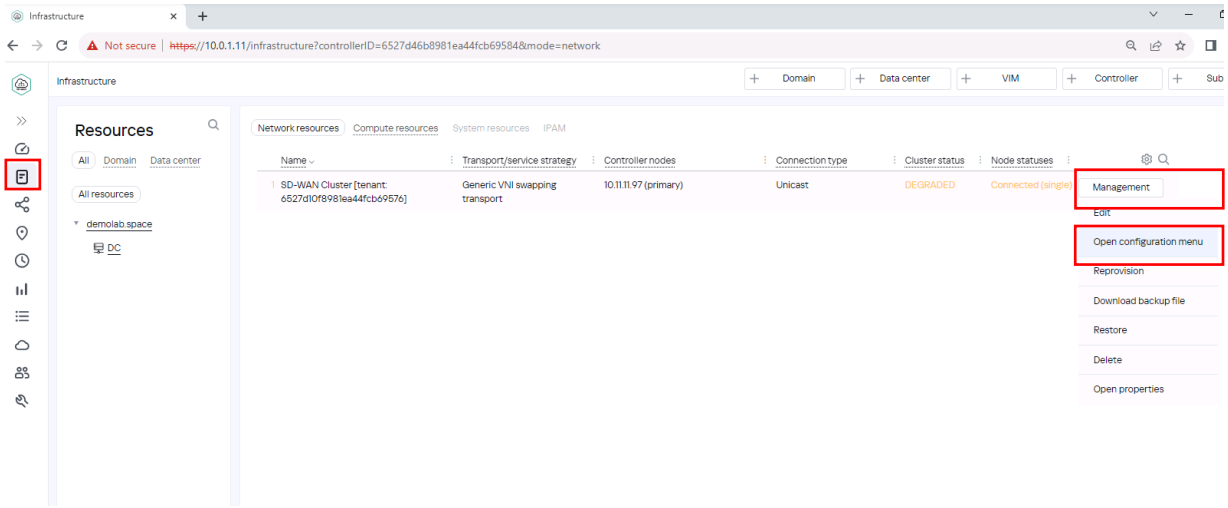
Отметить туннель как “ Unsolicited” – означает «нежелательный» для использования.

Нажать Save for both tunnels – сохранение параметров мониторинга туннелей в оба направления.


vCPE-3
Registered
Save
Close
Configuration
Monitoring
Problems
Encryption
Service requests
Tags
Scripts
SD-WAN settings
Topology
Network settings
BGP settings
OSPF
Routing Filters
More
Source
Destination
Unsolicited
Thresholds monitoring
MTU
Errors/second
Utilization (%)
Latency (ms.)
Jitter (ms.)
Packet loss (%)
Speed (MB/sec)
Cost
CPE [vGW-11: 8000005056AA9EA5] : 4800
CPE [vCPE-3: 8000005056AAC4FD] : 4800
CPE [vCPE-3: 8000005056AAC4FD] : 4801
CPE [vGW-12: 8000005056AAD2B1] : 4800
CPE [vGW-12: 8000005056AAD2B1] : 4801
Tunnel monitoring thresholds
Enable tunnel thresholds monitoring
Unsolicited
Interval for processing errors and utilization rate (sec)
60
Enable error monitoring
Critical error level (errors/sec)
1000
Enable utilization monitoring
Critical utilization level (%)
95
Interval for processing latency, jitter, and packet loss (sec)
15
Enable latency monitoring
Close
Save for both tunnels
Set to default
Save

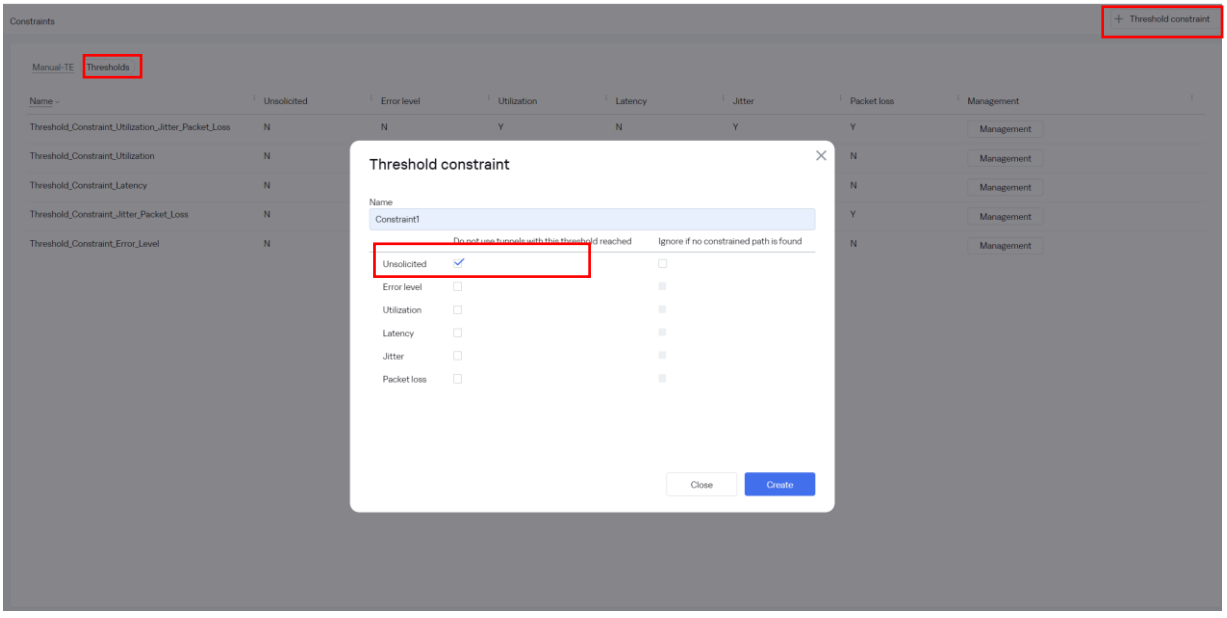
3.7.6. Создание Constraints.

Для перенаправления трафика необходимо создать ограничения. Для этого нужно перейти в меню *Infrastructure* > *SD-WAN контроллер* > *Management* > *Open configuration menu*.



Перейти в меню Constraints/ затем отрыть вкладку Thresholds и нажать на кнопку *+Threshold Constraint*.

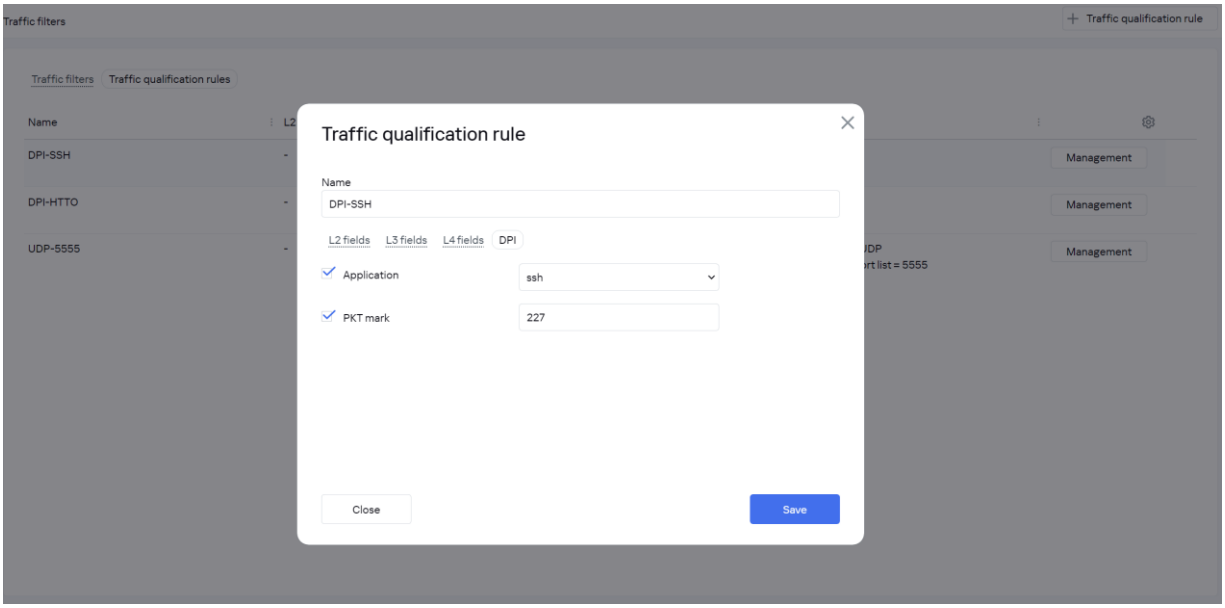
Задать название Constraints в поле name и включить ограничение Unsolicited Данное ограничение исключит из транспортного сервиса туннели, отмеченные как Unsolicited. Нажать Save.



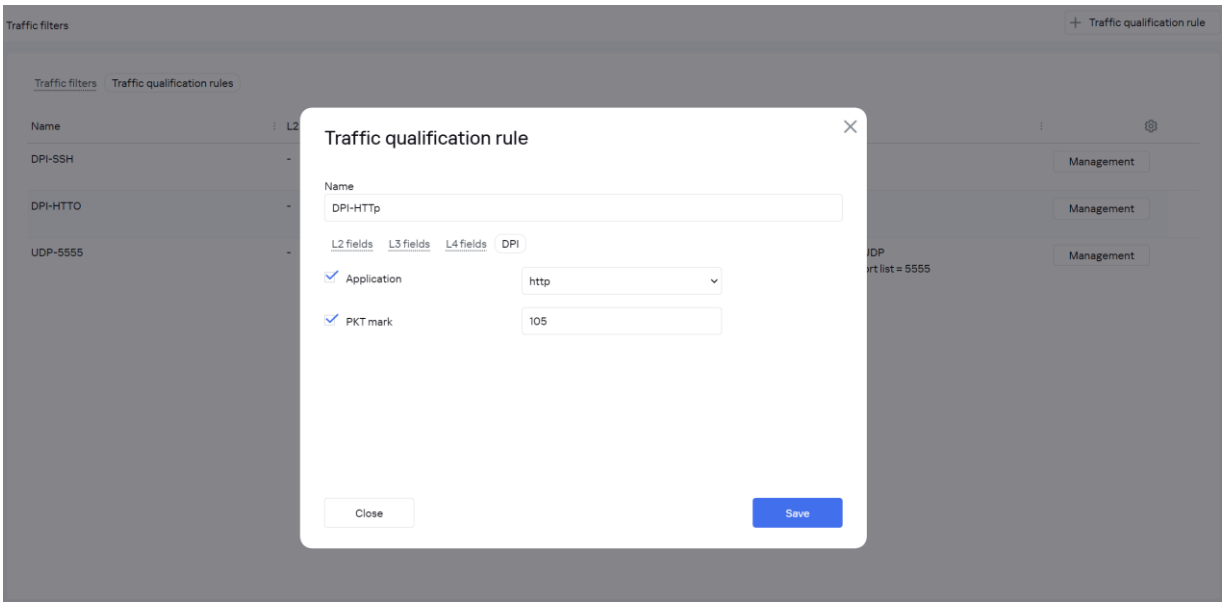
3.7.7. Создание правил фильтрации DPI.

Перейти в меню Traffic filters. Затем перейти во вкладку Traffic qualification rules. Нажать *+Traffic qualification rule*.

Задать имя правила, отметить во вкладке L3 Fields Protocol – IPv4, во DPI Application - SSH, и нажать Create.

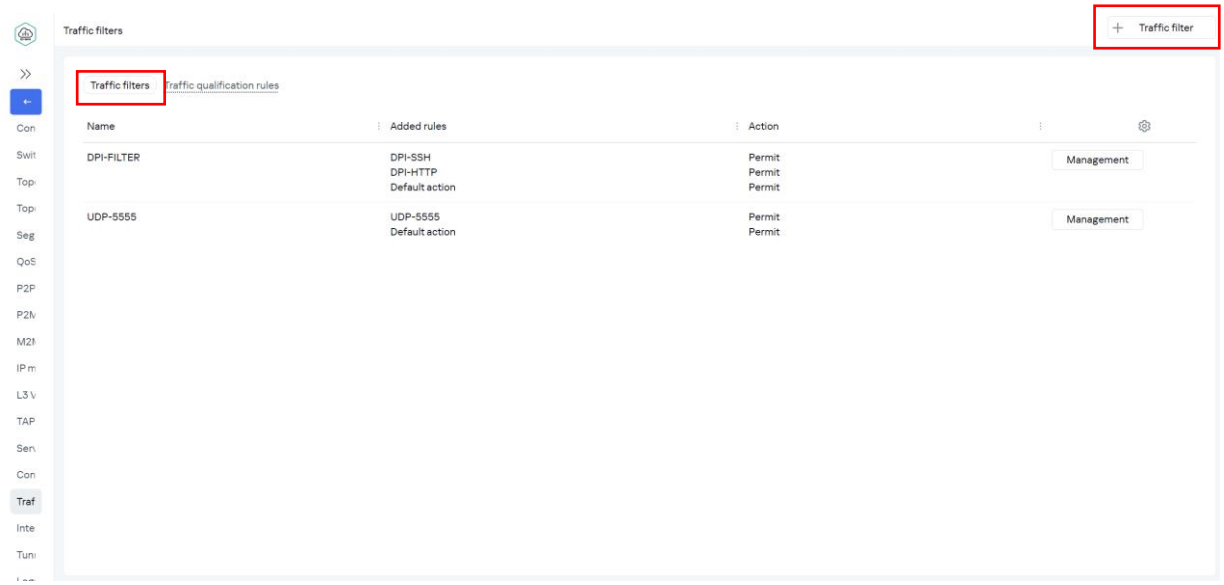


Для HTTP трафика аналогично задать имя правила, отметить во вкладке L3 Fields Protocol – IPv4, во DPI Application - http, и нажать Create.

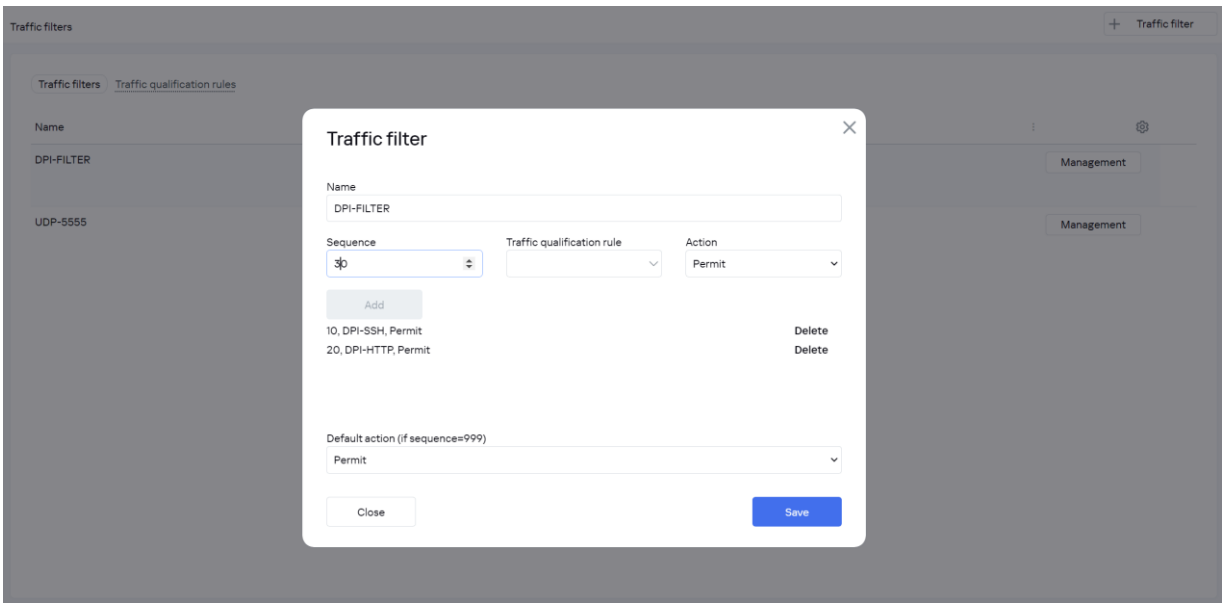


3.7.8. Создание фильтра.

Далее необходимо создать фильтр, куда добавятся созданные ранее правила. Перейти на вкладку Traffic filters, нажать +*Traffic Filter*.



- Задать имя,
- Добавить правила, созданные в п.3.7.7. Выбрать в селекторе Rule созданное п. 3.7.7 правило для SSH, Action – Permit. Нажать Add. Повторить для правила HTTP.
- нажать Create.



3.7.9. Создание ACL Service interfaces.

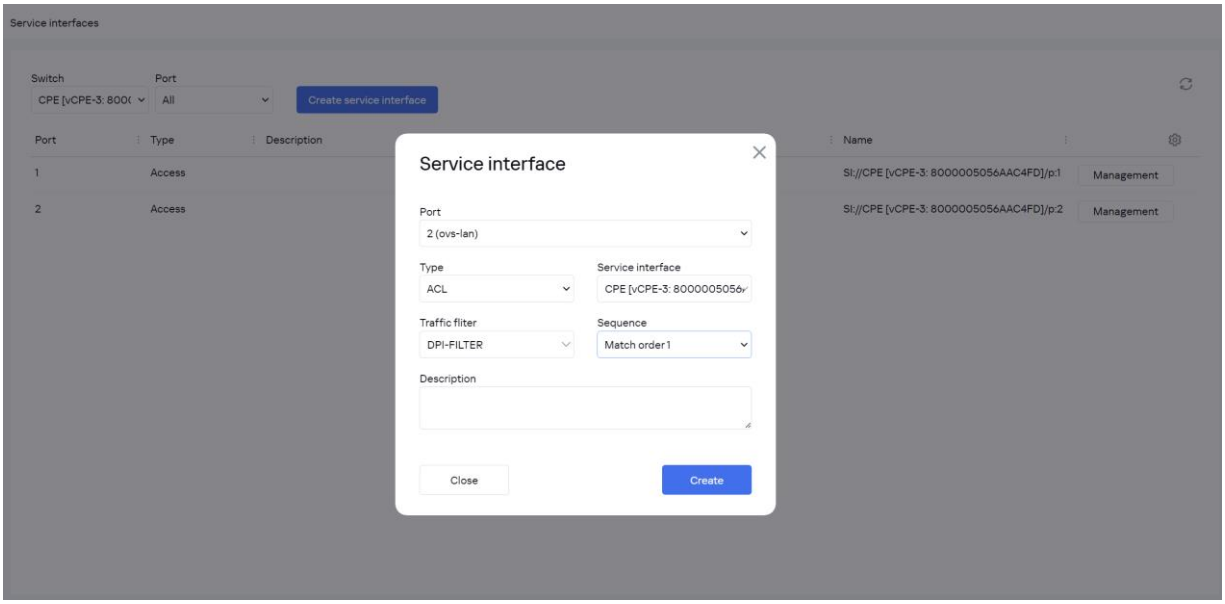
Трафик попадает в транспортный сервис через сервисные интерфейсы. Необходимо создать специальный ACL интерфейс (ACL Service Interfaces – ACL SI). Перейти на вкладку Service Interfaces, затем выбрать Switch vCPE-3 и Port 2 (ovs-lan).

Нажать Create service interface. Выбрать:

- Type: ACL.
- Service interface: vCPE-3.
- Traffic filter*: DPI-Filter*, созданный в пункте0.
- Sequence "*Match order 1"* (данный ACL SI будет с высшим приоритетом).

Нажать Save.

Повторить для vCPE-4.

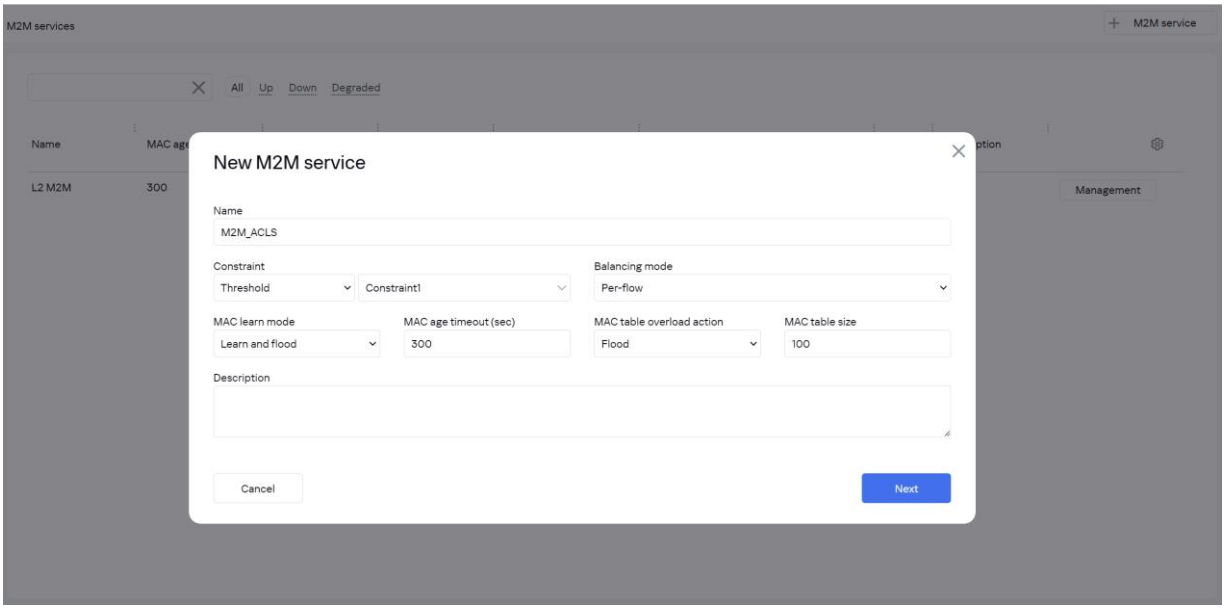


3.7.10. Для создания транспортного сервиса требуется создать ACL сервисные интерфейсы для каждой CPE. Повторить п.3.6.9 для vCPE-4.

3.7.11. Создание отдельного транспортного сервиса.

Перейти в M2M services. Нажать *+M2M service*.

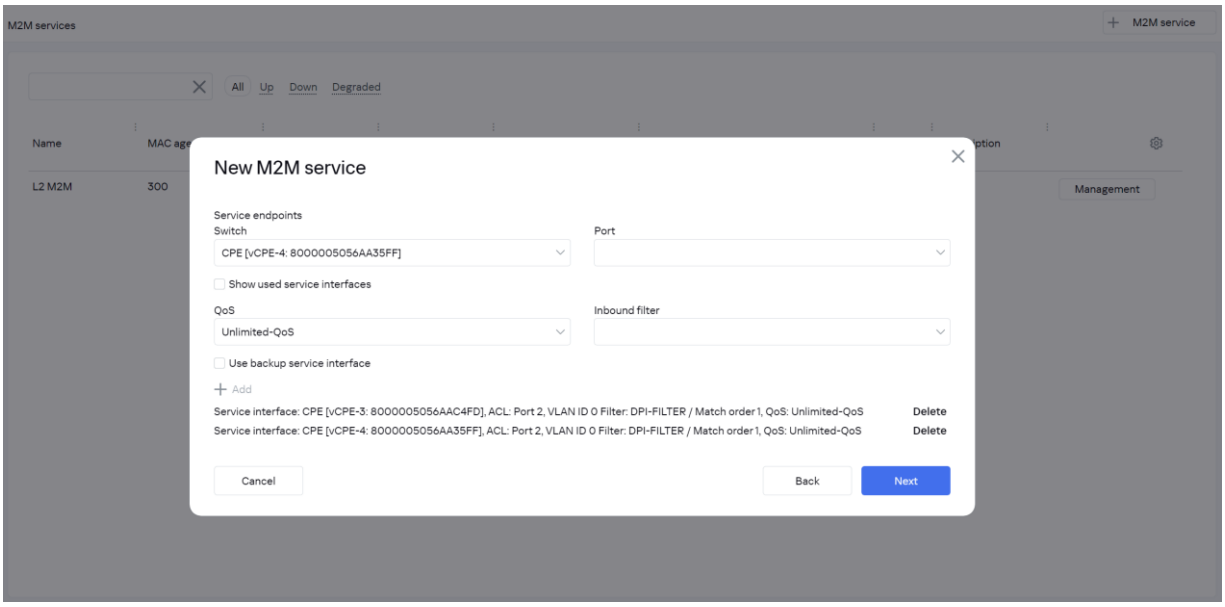
Задать название, выбрать созданный ранее Threshold (в пункте 3.6.6), нажать Next.



Добавить 2 сервисных интерфейса для направления трафика в сервис:

- выбрать Switch (vCPE3 и vCPE4)
- Port – созданные в п. 3.7.9 и 3.7.10 ACL SI
- Qos – Unlimited-QoS
- Нажать на +Add

Нажать Next и Save.



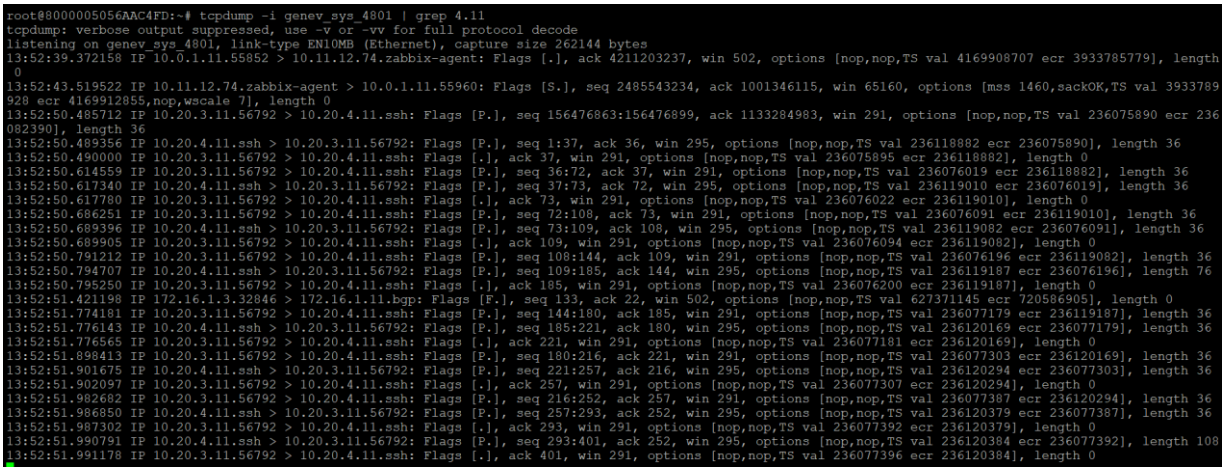
3.7.12. Подключиться к vCPE-3 и проверить, что трафик переключился на другой WAN интерфейс.

В пункте 3.7.2 проверялось, что трафик идёт через интерфейс genev_sys_4800 (sdwan0). После настройки отдельного транспортного сервиса, в результате работы ограничений и фильтра, трафик переключился на интерфейс genev_sys_4801 (sdwan1).

Проверить с помощью tcpdump наличие трафика на интерфейсе geneve_sys_4801.

 # tcpdump -i genev_sys_4801

На скриншоте видно, что трафик переключился с интерфейса genev_sys_4800 (sdwan0) на genev_sys_4801 (sdwan1).

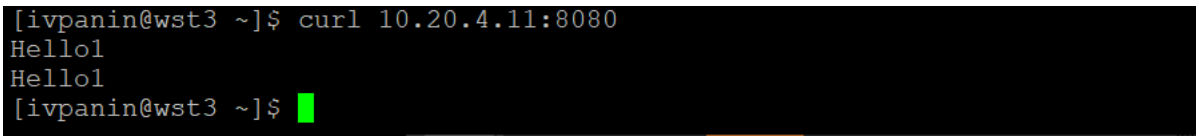


3.7.13. Для проверки HTTP возможно использовать nc на wst4:

[root@wst4]# echo Hello1 >> some.file
[root@wst4]# { printf 'HTTP/1.0 200 OK\r\nContent-Length: %d\r\n\r\n' "$(wc -c < some.file)"; cat some.file; } | nc -l 8080

Для генерации HTTP запроса открыть с wst3 HTTP сессию на порт 8080 wst4. Например, с помощью curl:

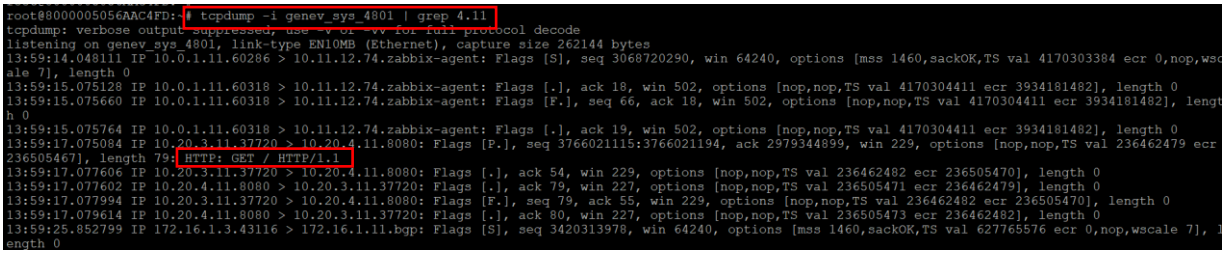
[root@wst3]# curl 10.20.4.11:8080



Затем подключиться к vCPE-3 и проверить, что трафик переключился на корректный интерфейс (в зависимости от настроек, сделанных ранее):

 # tcpdump -i genev_sys_4801

На примере ниже видно, что HTTP трафик переключился с интерфейса genev_sys_4800(WAN0) на 4801 и DPI распознал HTTP трафик на нестандартном порту.



3.7.14. Возврат настроек после завершения теста.

Удалить сервис, созданный в п. 3.7.11 (при удалении отметить Remove associated service interfaces).

Убрать параметр “Unsolicited” с туннелей, добавленный в п.3.7.5.

4. **Построение топологии SD-WAN сети.**

В решении Kaspersky SD-WAN возможны следующие варианты топологий:

- Hub-and-Spoke. Топология по умолчанию, которая используется в том случае, если устройствам CPE не назначено топологических тегов. Такие устройства не устанавливают прямые туннели между собой, весь трафик в этом случае идет через шлюз SD-WAN.
- Full-Mesh. Для построения данной топологии необходимо назначить устройствам CPE одинаковый топологический тег для реализации этой топологии. Все устройства с одинаковым топологическим тегом устанавливают прямые туннели между собой.
- Partial-Mesh. Возможно, группировать устройства CPE путем назначения одного топологического тега одной группе устройств и другого топологического тега другой группе. В этом случае все устройства CPE из одной группы (с одинаковым топологическим тегом) пытаются установить прямые туннели между собой, а с устройствами из другой группы взаимодействуют через шлюз.

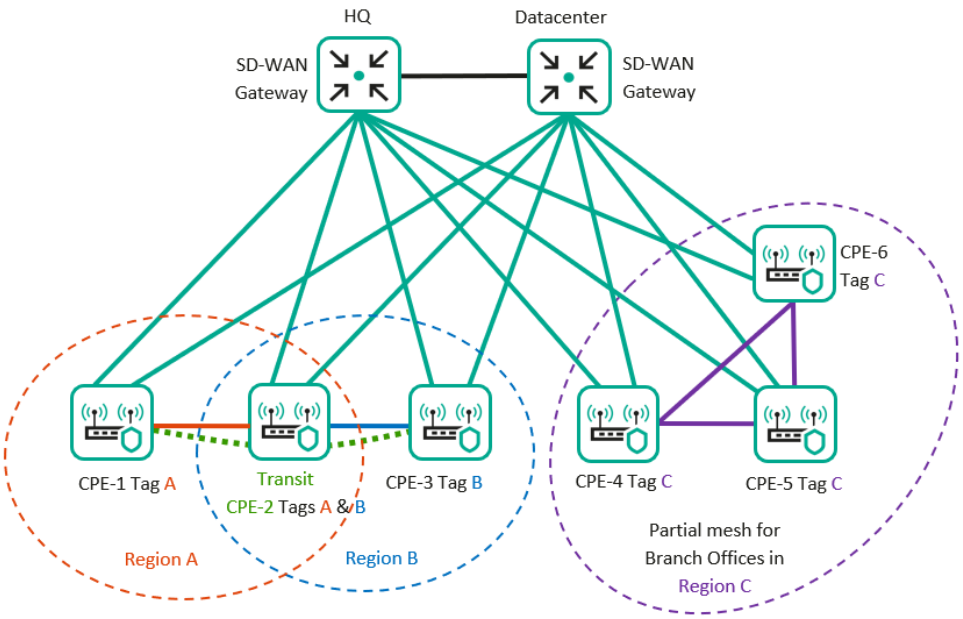


Рис. 5.1 Варианты топологий SD-WAN сети.

Для построения сетевых топологий в решении Kaspersky SD-WAN используются топологические теги, которые назначаются устройствам CPE.

Также устройство CPE может быть транзитным. В этом случае другие устройства CPE могут устанавливать через него туннели.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Настройка топологии:

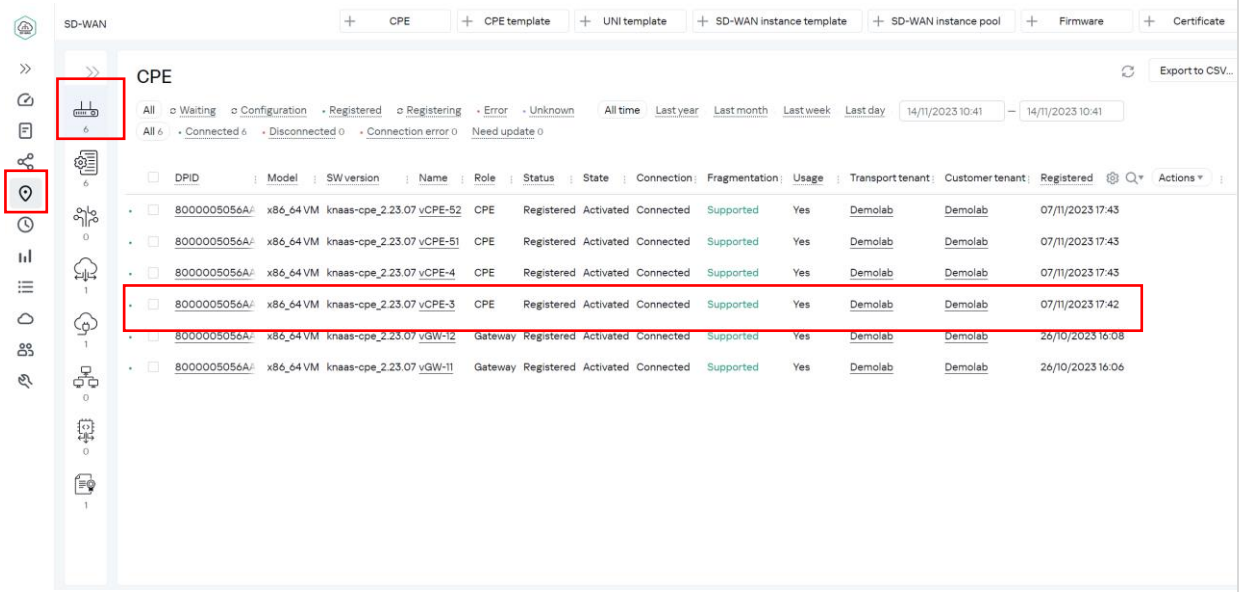
https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/250942.htm

4.1. **Создание топологий Full-Mesh.**

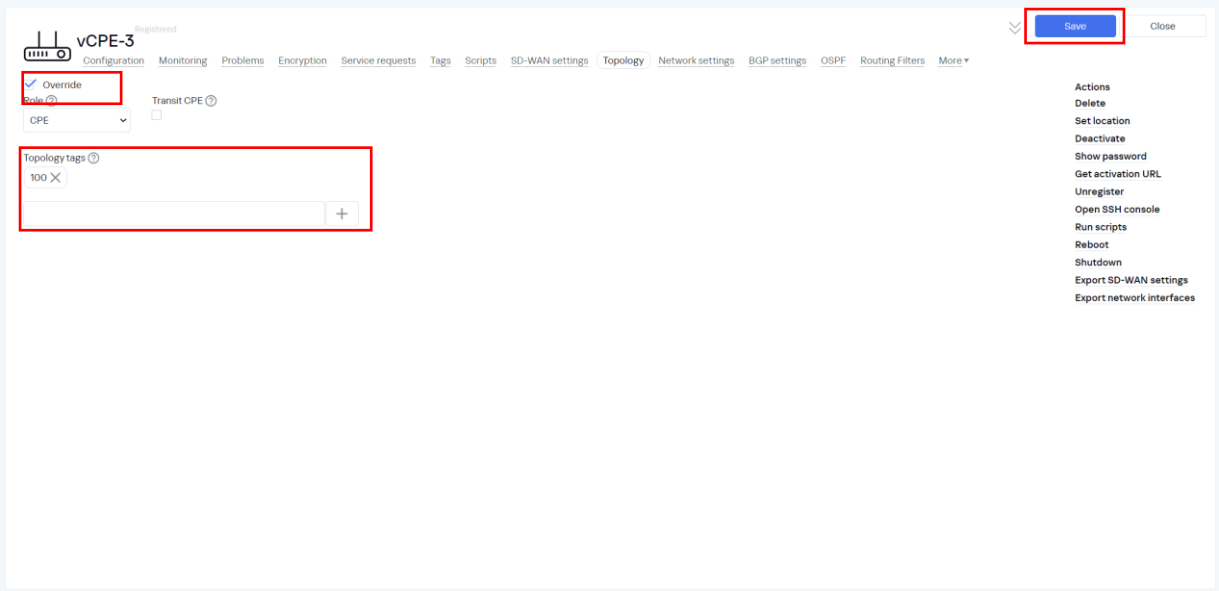
В данном сценарии настраивается топологии Full-Mesh между устройствами CPE, для этого будет добавлен одинаковый топологический тег для устройств CPE. Построенная топология будет отображена в настройках транспортного сервиса. Также будут отображены дополнительно построенные пути между устройствами CPE в разделе Сегменты.

4.1.1. Настройка топологических тегов.

Для создания топологии Full-Mesh устройства CPE должны иметь одинаковые топологические теги.
Для настройки перейти в меню CPE и выбрать vCPE-3.

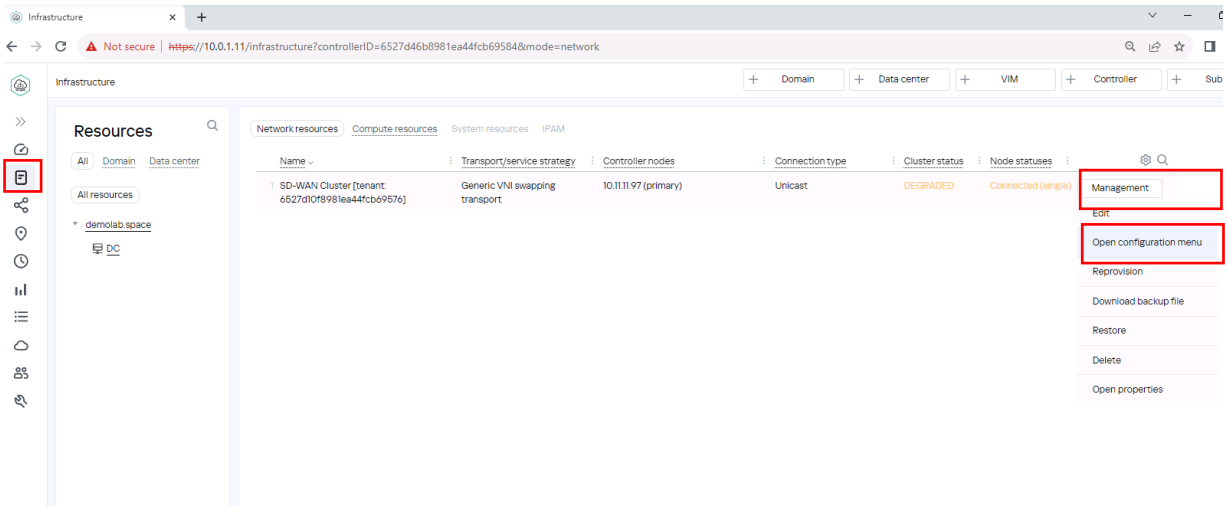


Перейти на вкладку *Topology*. Отметить *Override* и добавить тег 100 (нажать на +).
Нажать Save (оркестратор применит измененные настройки к CPE).

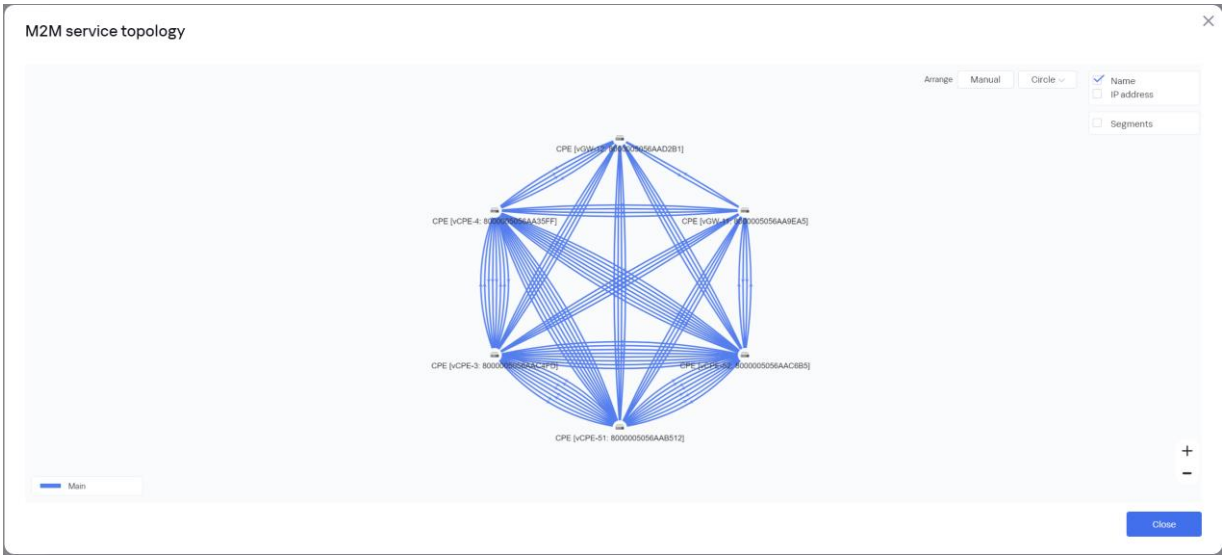


4.1.2. Далее необходимо назначить топологический тег 100 для остальных устройств CPE. Повторить пункт 4.1.1 для устройств vCPE-4, vCPE-51, vCPE-52.

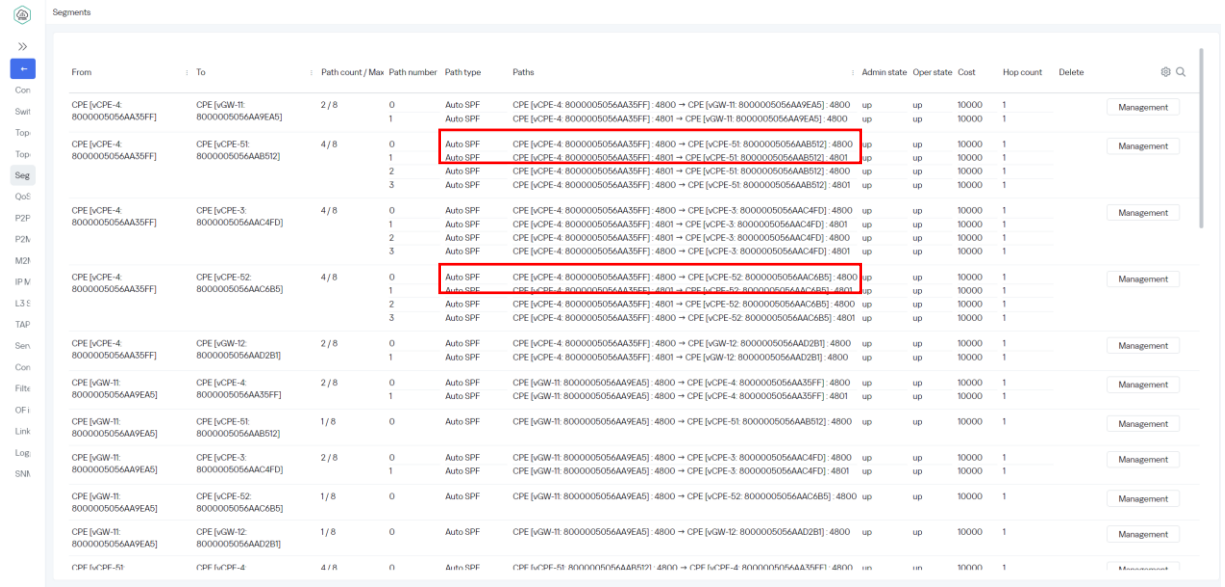
4.1.3. Для просмотра построенной топологии перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



4.1.4. Выбрать для сервиса L2 M2M меню Management > View service topology. Отобразится построенная топология сервиса. На скриншоте представлена Full-Mesh топология между CPE, также устройства CPE сохранили туннели до vGW(шлюзов). Для удобства просмотра топологии можно выбрать Arrange – Circle и отметить Name.



4.1.5. Для проверки построенных путей между устройствами CPE перейти на вкладку Segments. Представлен список сегментов, где видны построенные сегменты. На скриншоте ниже видно, что построены сегменты между устройствами vCPE, не проходящие через шлюзы(vGW).
Таким образом vCPE сформировали Full-Mesh между собой.



4.1.6. Возврат настроек после завершения теста.

Убрать теги с CPE устройств, добавленные в п. 4.1.1 и 4.1.2.

4.2. **Создание топологий Partial-Mesh.**

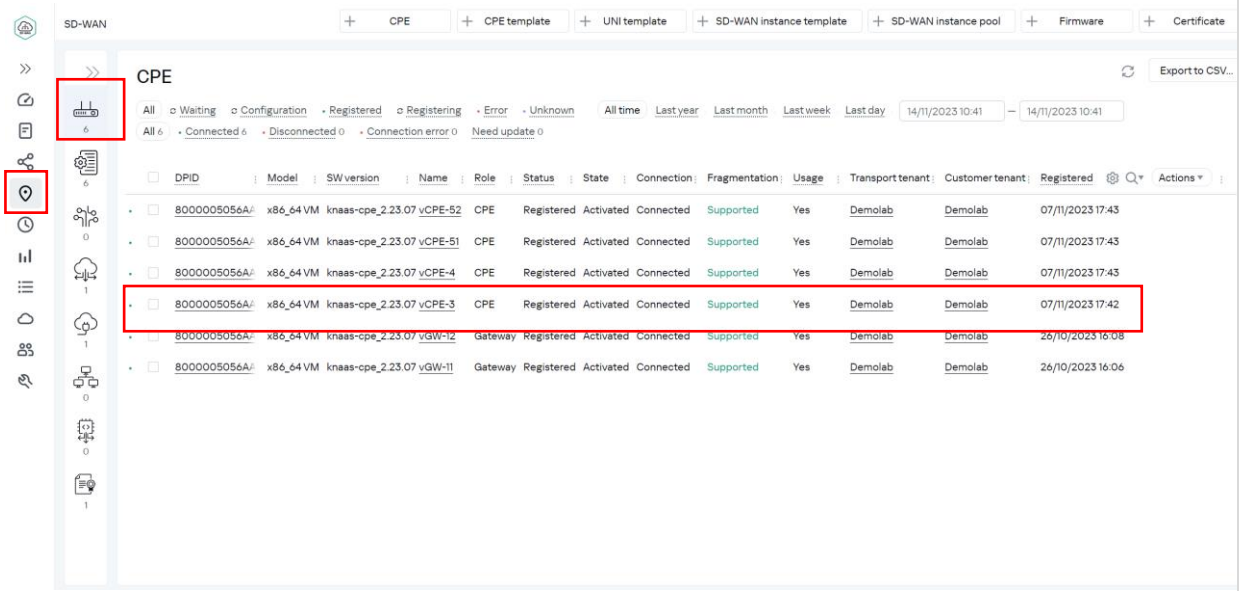
В данном сценарии настраивается топологии Partial-Mesh между устройствами CPE. Будут сформированы 2 группы устройств CPE:

- vCPE-3 и vCPE-4
- vCPE-51, vCPE-52 и vCPE-4

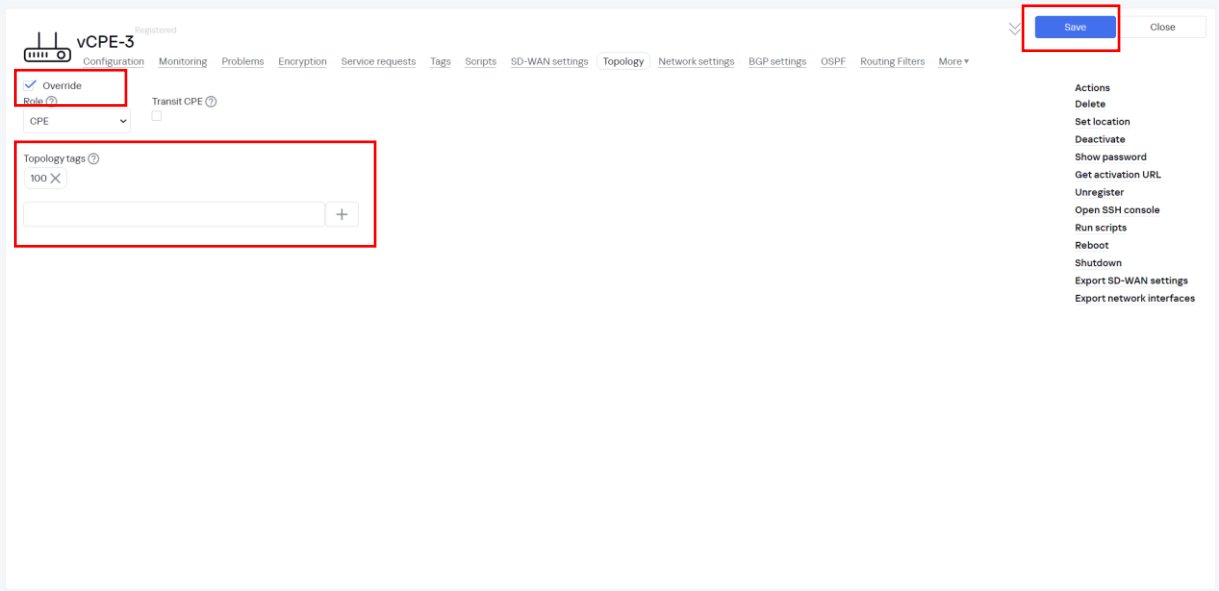
Для построения топологии Partial-Mesh будут назначены топологические теги для устройств CPE, отдельно для каждой группы. Построенная топология будет отображена в настройках транспортного сервиса. Также будут видны дополнительно построенные пути между устройствами CPE.

4.2.1. Для создания топологии Partial-Mesh необходимо назначить устройствам CPE различные топологические теги в соответствии с требуемой топологией.

Необходимо перейти в меню CPE и выбрать vCPE-3.



4.2.1. Перейти на вкладку Topology. Отметить Override и добавить тег 100 (нажать на +). Нажать Save (оркестратор применит измененные настройки к CPE).

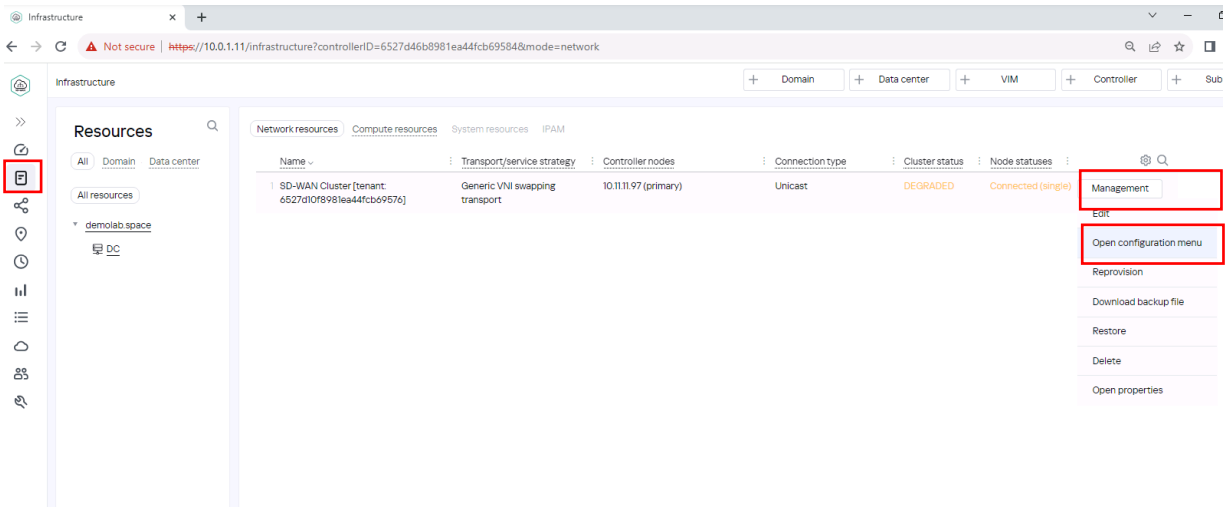


4.2.2. Для топологии Partial-Mesh необходимо назначить разные теги группам CPE. В сценарии создается 2 группы CPE: vCPE3 и vCPE-4 с тегом 100 и vCPE4, vCPE-51, vCPE-52 с тегом 200. Для их назначения повторить пункты 4.2.1-4.2.2 для остальных CPE со следующими значениями тегов:

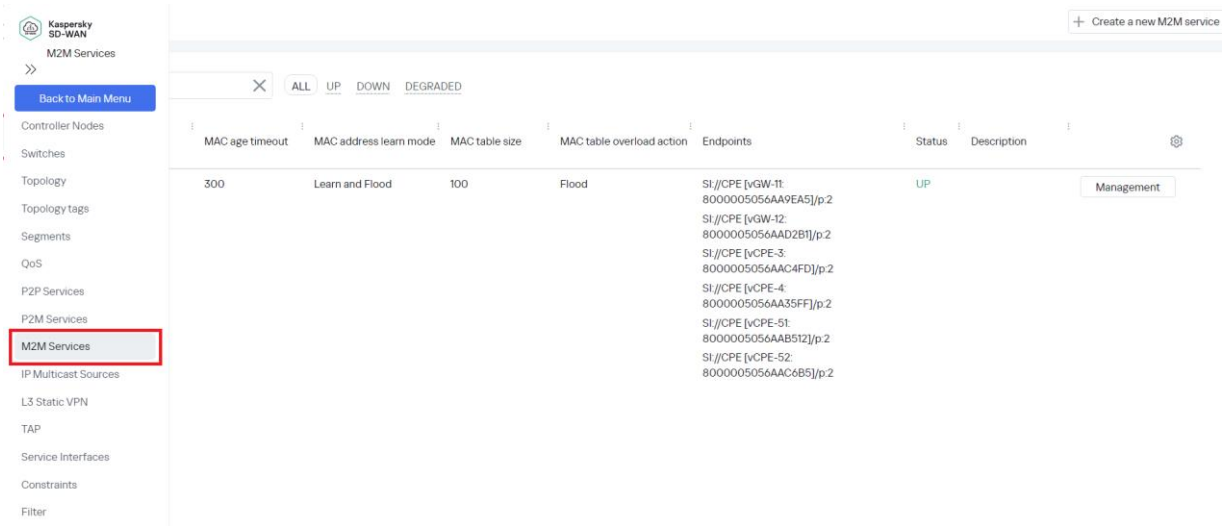
- vCPE-51 – 200.
- vCPE-52 – 200.
- vCPE-4 – 100 и 200.

4.2.3. Просмотр сформированной топологии.

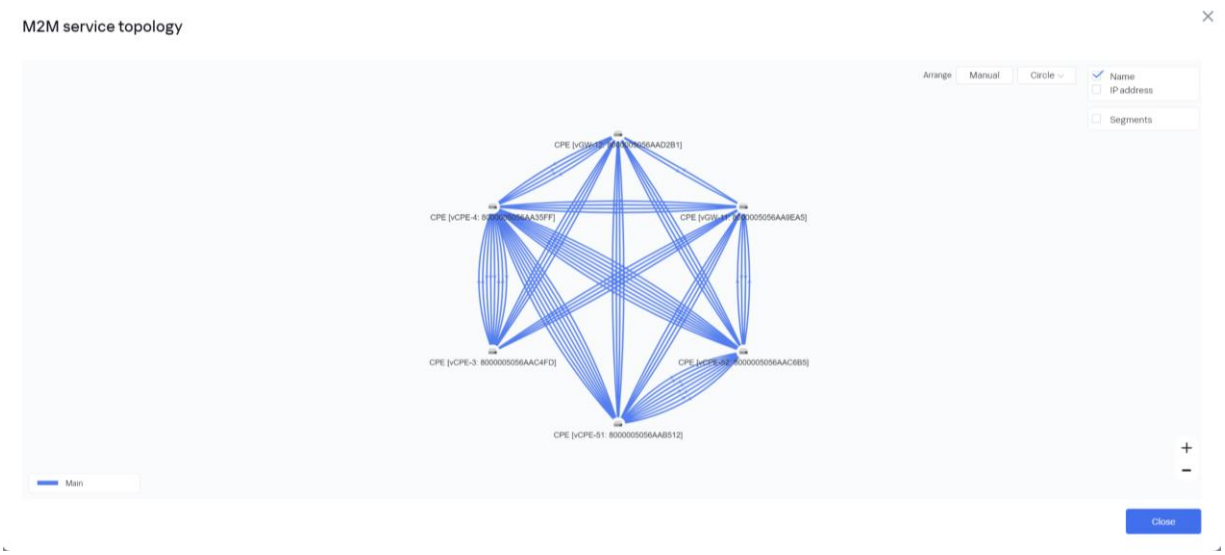
Перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



Открыть M2M Services.

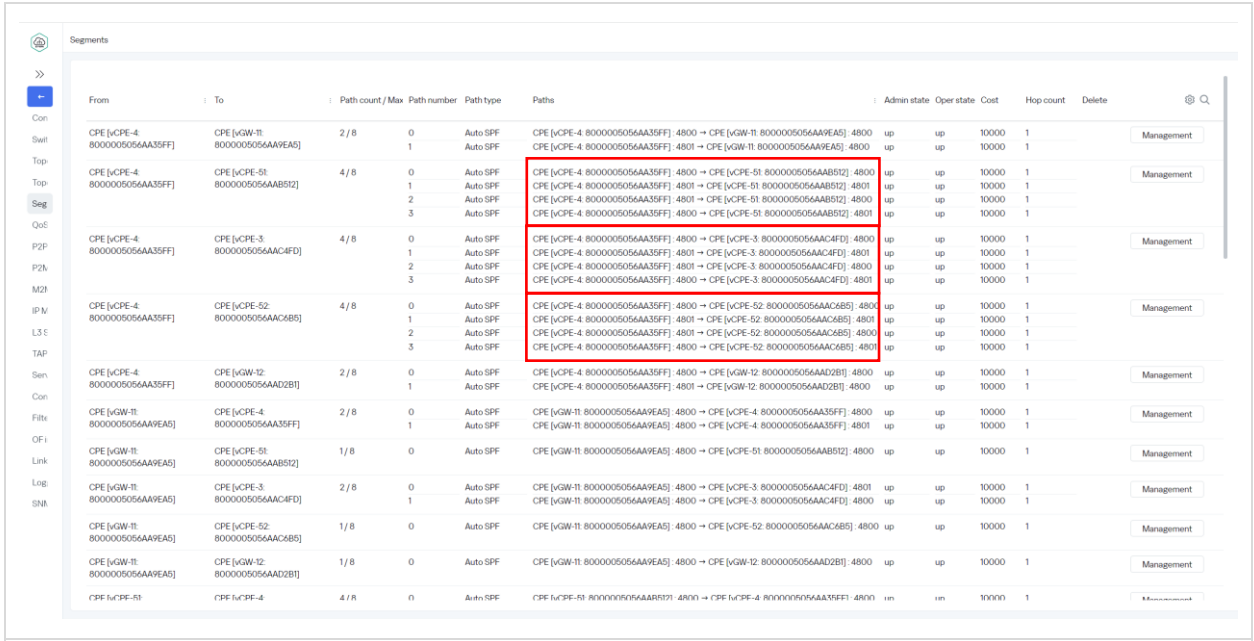


Выбрать для сервиса *L2 M2M* меню Management > View service topology.
На скриншоте отображено, что устройства CPE построили туннели между CPE-3 и CPE-4, Full-Mesh между CPE-4, CPE-51 и CPE-52, а также сохранили туннели до vGW. Для удобства просмотра топологии можно выбрать Arrange – Circle и отметить Name.



4.2.4. Проверка построенных путей между устройствами CPE.

Перейти в раздел Segments. Отобразится список сегментов, где будут видны построенные пути между устройствами CPE. Видно, что построенные сегменты образуют Partial-Mesh топологию в соответствии с настроенными тегами (построены сегменты между vCPE-4 и vCPE-51/52. Но не между vCPE-3 и vCPE-51/52).



4.2.5. Возврат настроек после завершения теста.

Убрать теги с CPE устройств, добавленные в п. 4.2.1 и 4.2.2.

4.3. **Создание топологий с использованием транзитных CPE.**

Устройства CPE также могут быть транзитными, в таком случае через них могут строится сегменты между другими CPE. В данном сценарии для демонстрации работы функционала транзитных CPE будет использоваться топология Partial-Mesh.

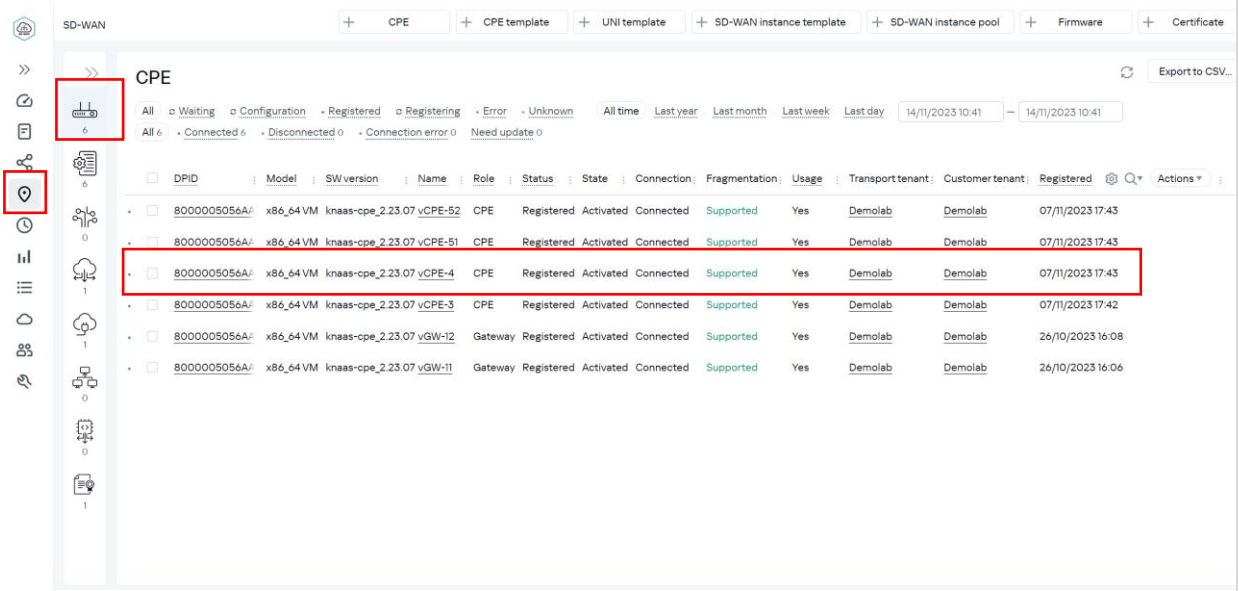
Будут сформированы 2 группы устройств CPE:

- vCPE-3 и vCPE-4.
- vCPE-4, vCPE-51, vCPE-52.

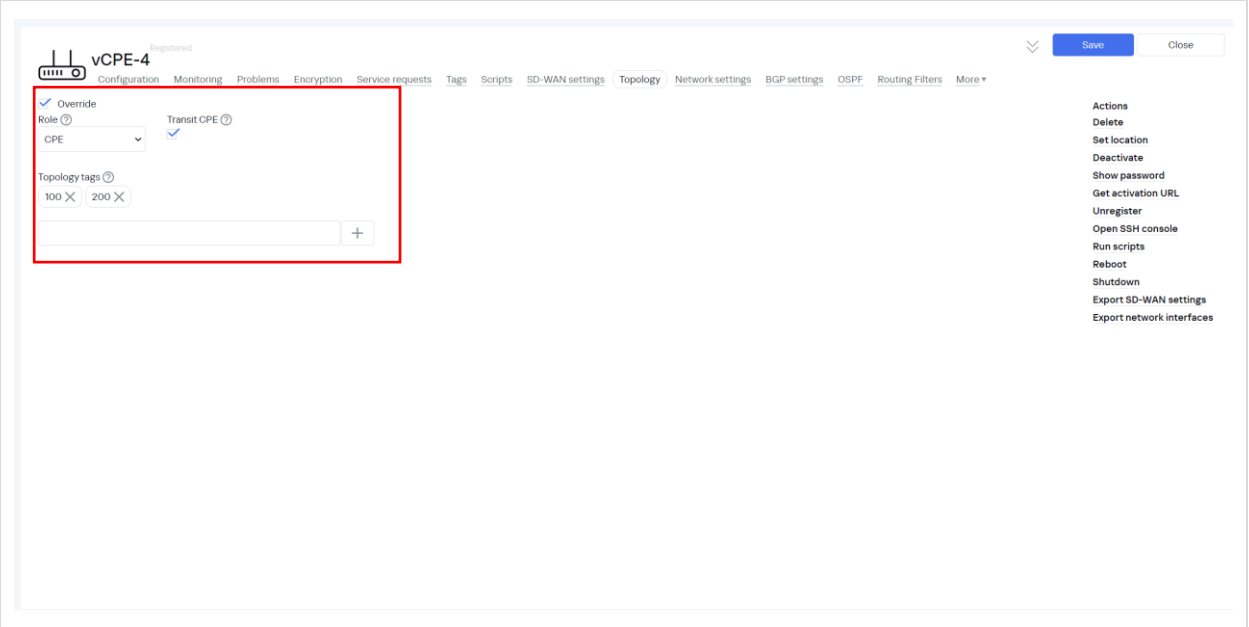
Каждой группе устройств CPE, будут назначены собственные топологические теги. Устройству vCPE-4 будет назначена транзитная роль, что позволит другим CPE строить туннели через данное устройство. Построенная топология будет отображена в настройках транспортного сервиса.

4.3.1. Настройка топологических тегов.

Для настройки тегов и транзитной роли необходимо перейти в меню CPE и выбрать vCPE-4.



4.3.2. Перейти в раздел Topology. Отметить Override, Transit CPE и добавить теги 100 и 200 (нажать на +). vCPE-4 выполняет транзитную роль и, благодаря этой настройке туннели между другими CPE будут также проходить через vCPE-4, а не только через шлюзы (vGW-11 и vGW-12). Нажать Update configuration (оркестратор применит новые настройки к CPE).

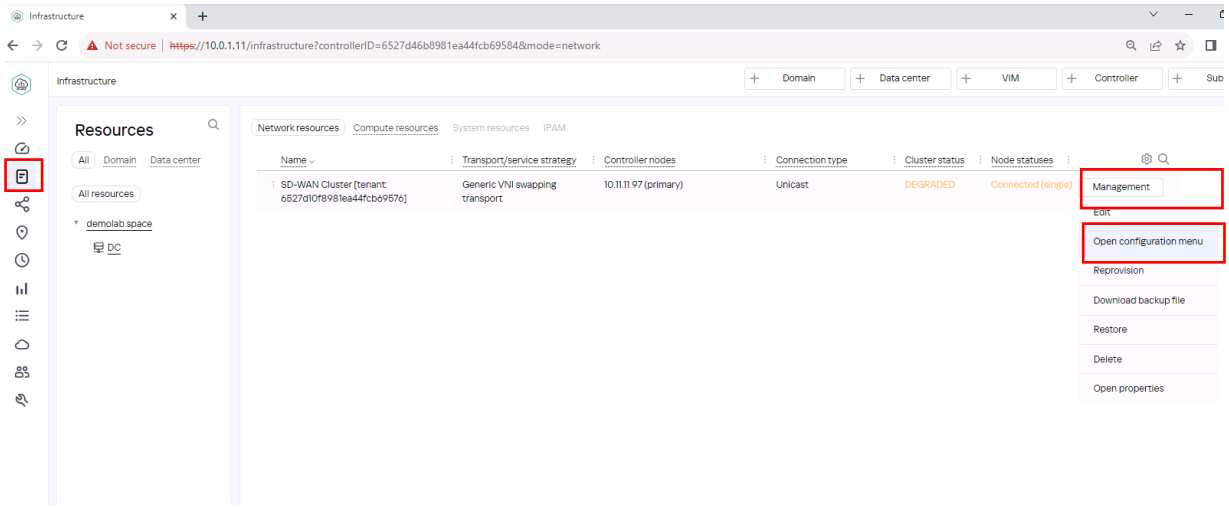


4.3.3. Для топологии необходимо назначить разные теги разным группам CPE. В сценарии создается 2 группы CPE: vCPE3 и vCPE-4 с тегом 100 и vCPE4, vCPE-51, vCPE-52 с тегом 200. Для назначения тегов пункты 4.3.1 - 4.3.2 для остальных CPE со следующими значениями тегов (эти CPE не будут транзитными для них не требуется отмечать Transit CPE):

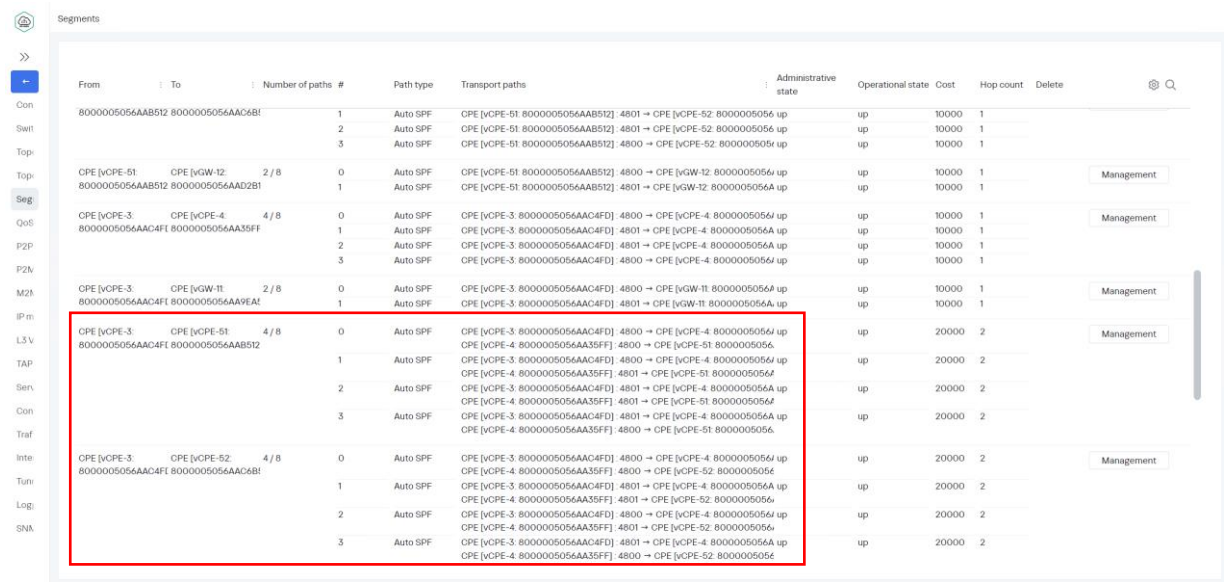
- vCPE-3 – 100.
- vCPE-51 – 200.
- vCPE-52 – 200.

4.3.4. Проверка построенной топологии.

Перейти в меню Infrastructure > SD-WAN контроллер > Management > Open configuration menu.



Открыть вкладку Segments. На скриншоте видно, что часть сегментов построена через vCPE-4, т.к. устройство было настроено как транзитное.



4.3.5. Возврат настроек после завершения теста.

Убрать теги с CPE устройств, добавленные в п. 4.3.1-4.3.3.

5. **Работа с CPE устройствами.**

5.1. **Централизованное обновление firmware CPE устройств.**

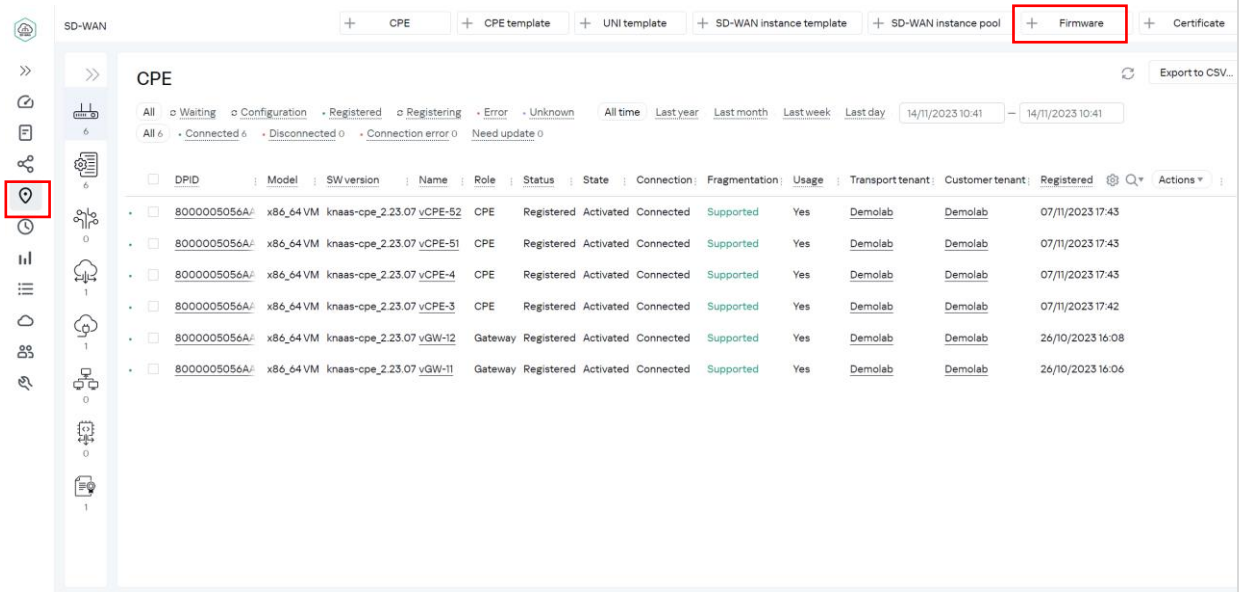
Kaspersky SD-WAN поддерживает централизованное обновление firmware («прошивок») на устройствах CPE. Перед установкой новой версии прошивки ее необходимо загрузить через веб-интерфейс оркестратора. Прошивки распространяются в виде архивов в формате TAR.GZ. Каждый архив содержит саму прошивку, а также файл с метаданными в формате yml. Параметры, указанные в файле с метаданными, импортируются в веб-интерфейс оркестратора при добавлении архива с прошивкой.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Прошивки: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/247435.htm

В данном сценарии рассматривается централизованное обновление firmware CPE.

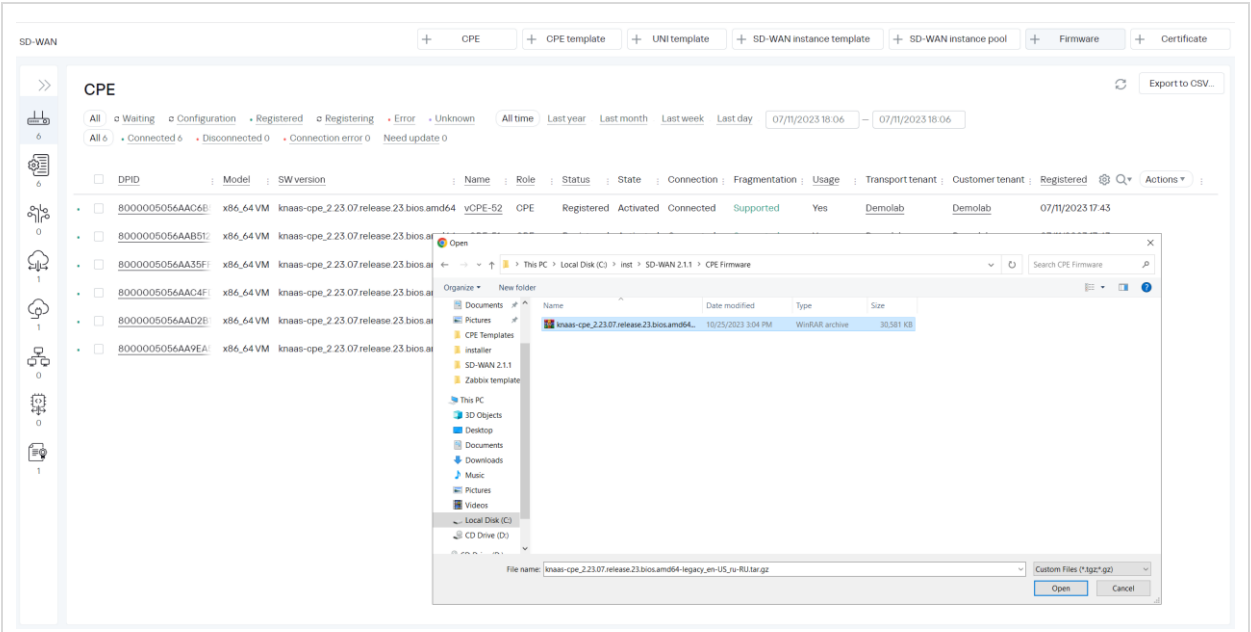
5.1.1. Загрузка новой версии firmware.

Для загрузки новой версии firmware необходимо перейти в меню SD-WAN.



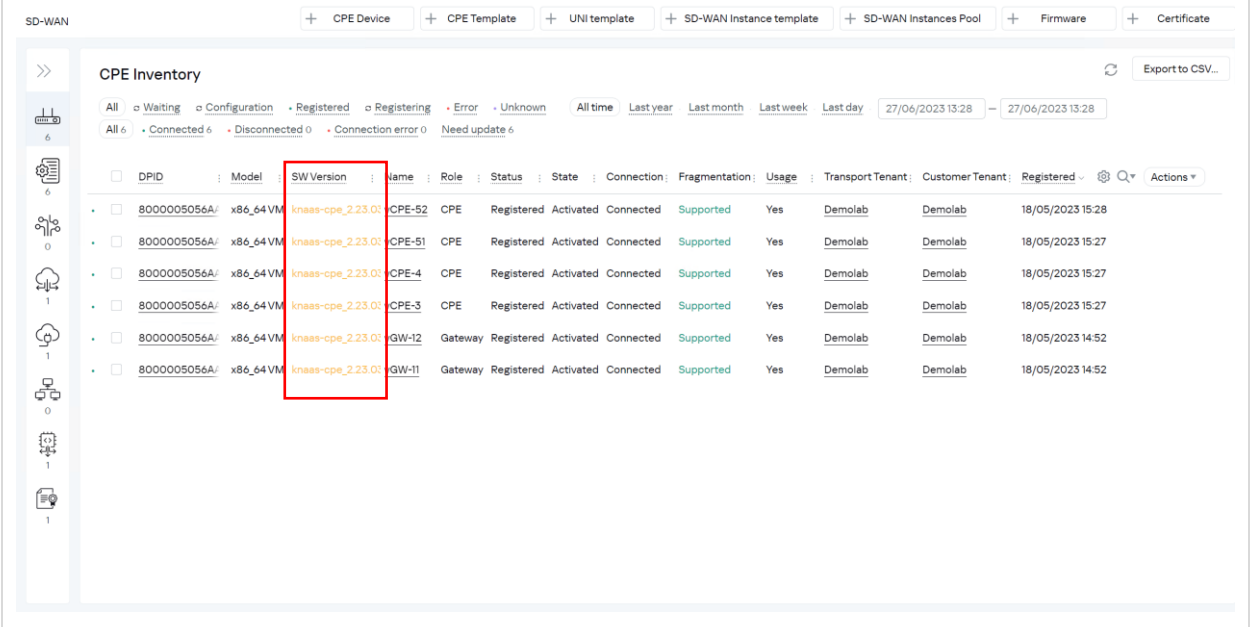
Нажать на кнопку *+Firmware* и в диалоговом окне загрузить файл с новой версией ПО для CPE.

Дождаться окончания загрузки (статус будет отображаться вместо кнопки *+Firmware*).



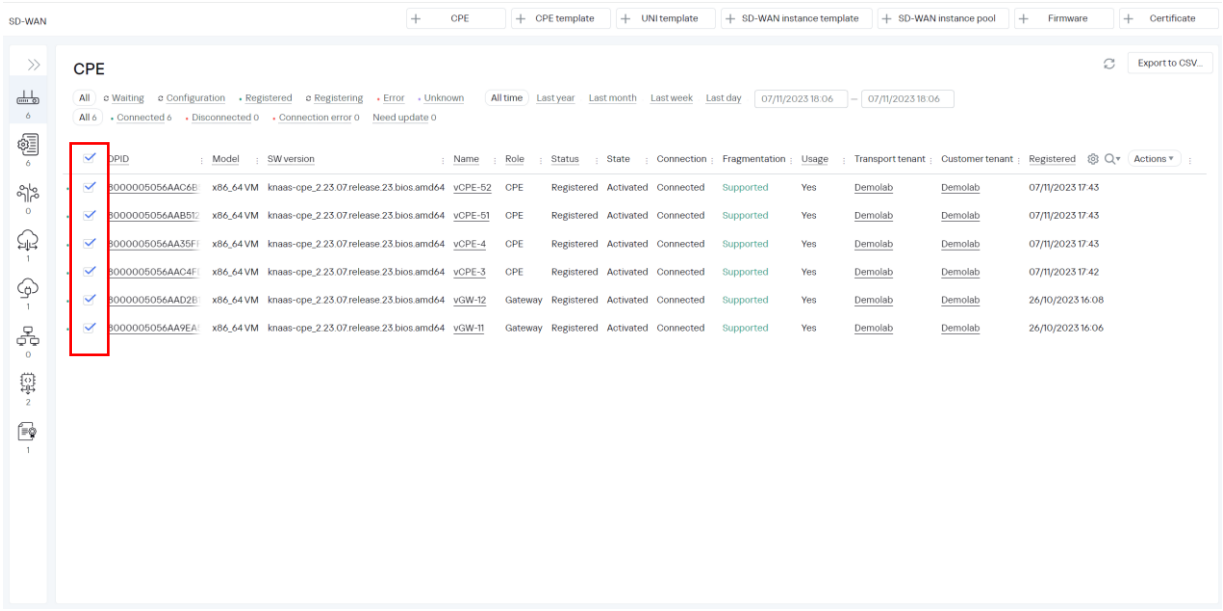
5.1.2. Просмотр CPE, доступных для обновления.

Перейти в меню CPE.
Откроется список CPE.
CPE, для которых доступна новая версия ПО, будут выделены цветом в колонке SW version. Новая версия firmware доступна для всех CPE.

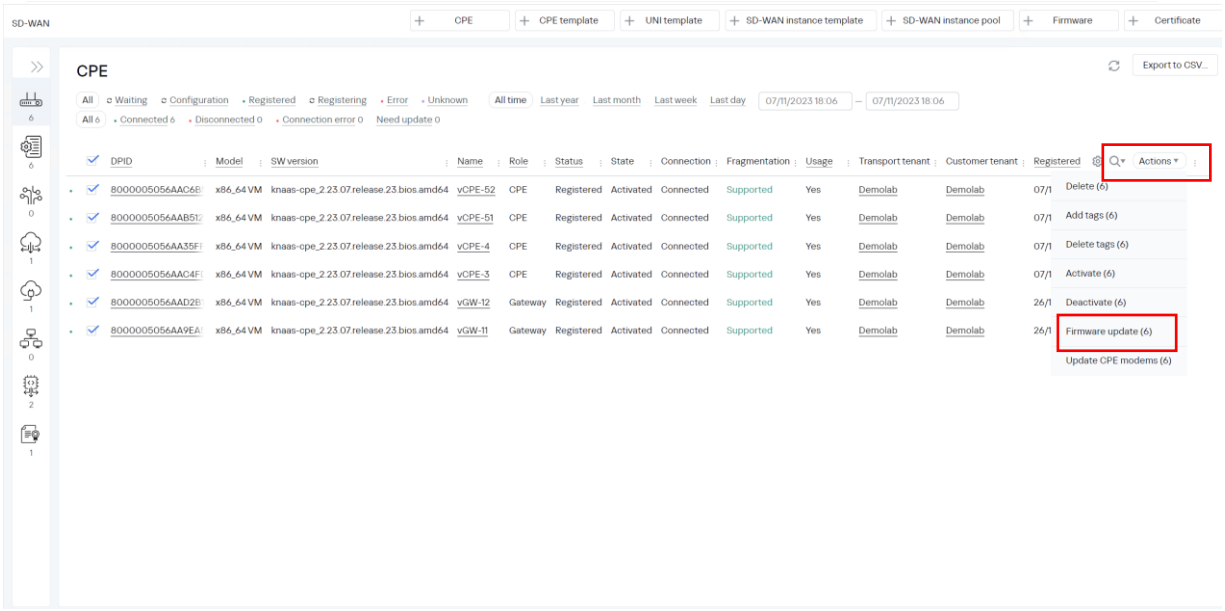


5.1.3. Создание задания (task) на обновление firmware.

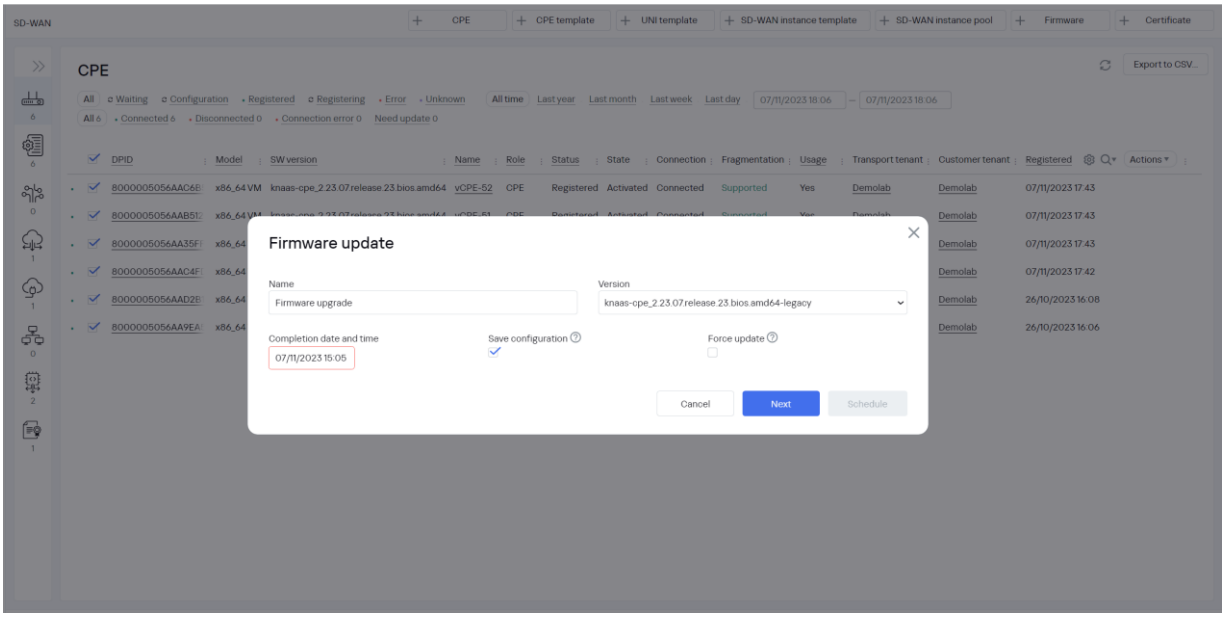
В CPE Inventory отметить слева все CPE, для которых требуется обновить прошивку.



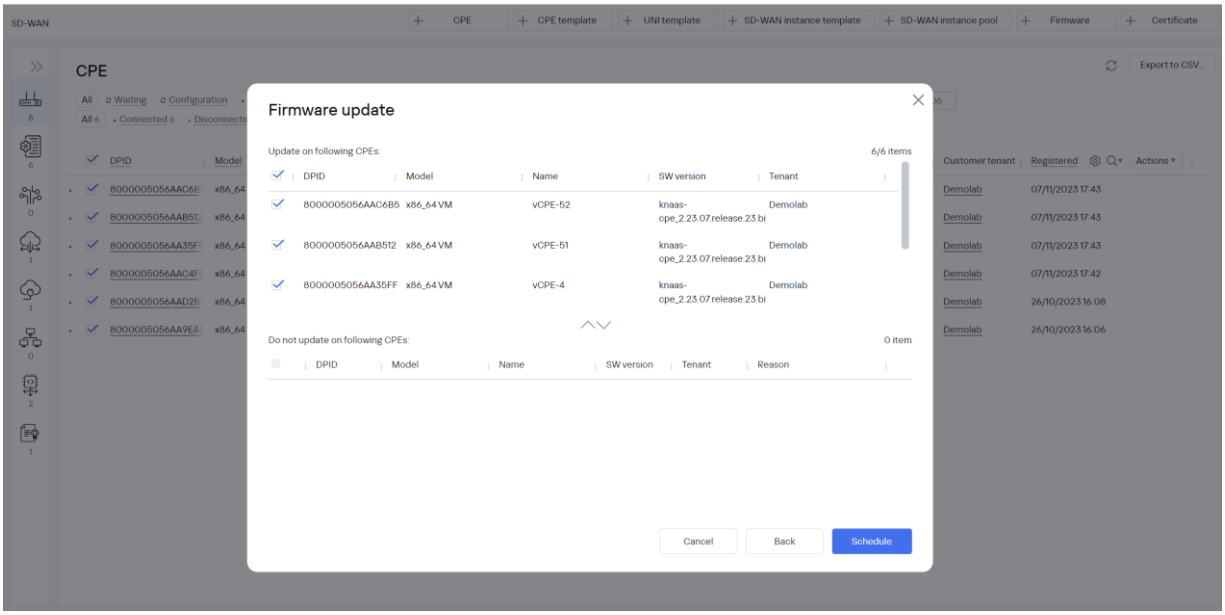
Справа в меню выбрать Actions > Firmware update.



Отобразится меню планировщика обновления. Далее необходимо задать имя задачи, версию, на которую требуется обновить CPE и время начала выполнения. Нажать Next.

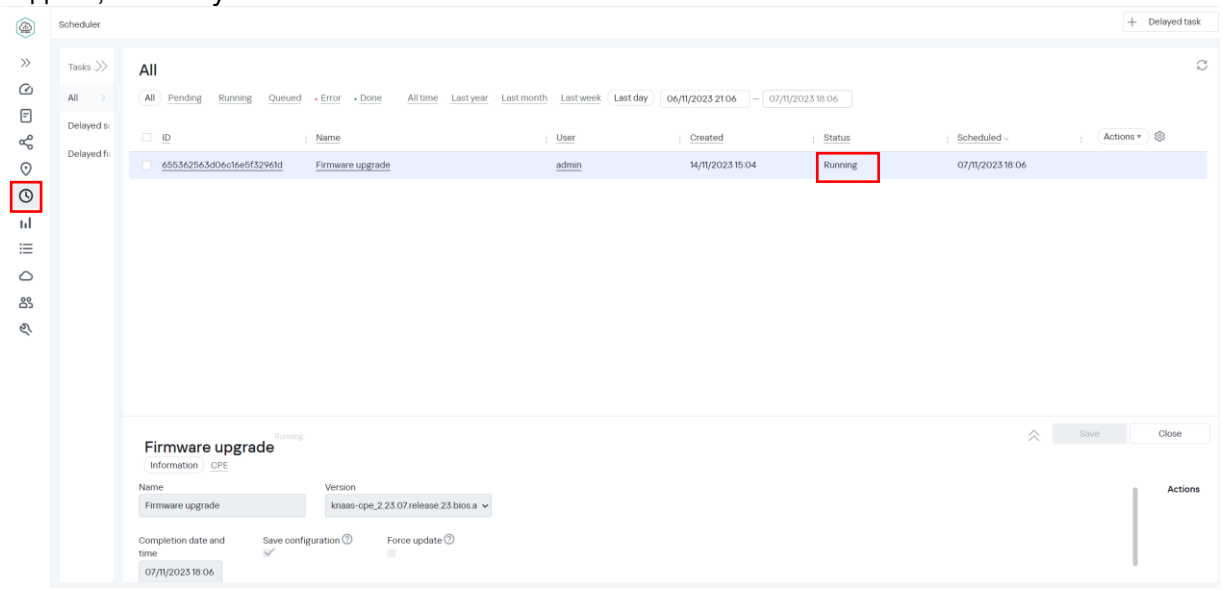


Отметить CPE для обновления и нажать Schedule.



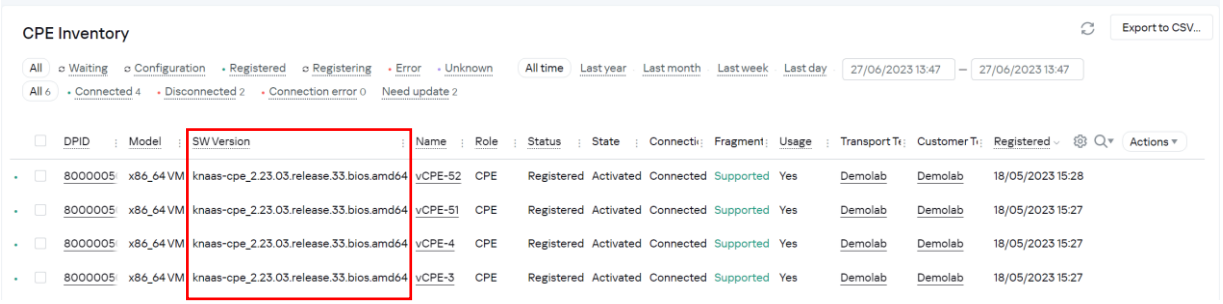
5.1.4. Проверка статуса задачи

Перейти в меню Scheduler.
Найти в списке созданную задачу и посмотреть статус. После успешного выполнения задачи, её статус изменится на Done.



5.1.5. Просмотр версий firmware CPE после обновления

Перейти в меню CPE.



Отобразится список CPE, в колонке SW Version будет указана новая версия прошивки, также, значение версии не будет выделено цветом.

Обновление успешно завершено.

5.2. **Резервирование устройств CPE с использованием VRRP.**

Kaspersky SD-WAN поддерживает установку нескольких устройств CPE на площадках для обеспечения высокой доступности. Одним из вариантов организации высокой доступности является использование протокола VRRP (Virtual Router Redundancy Protocol).

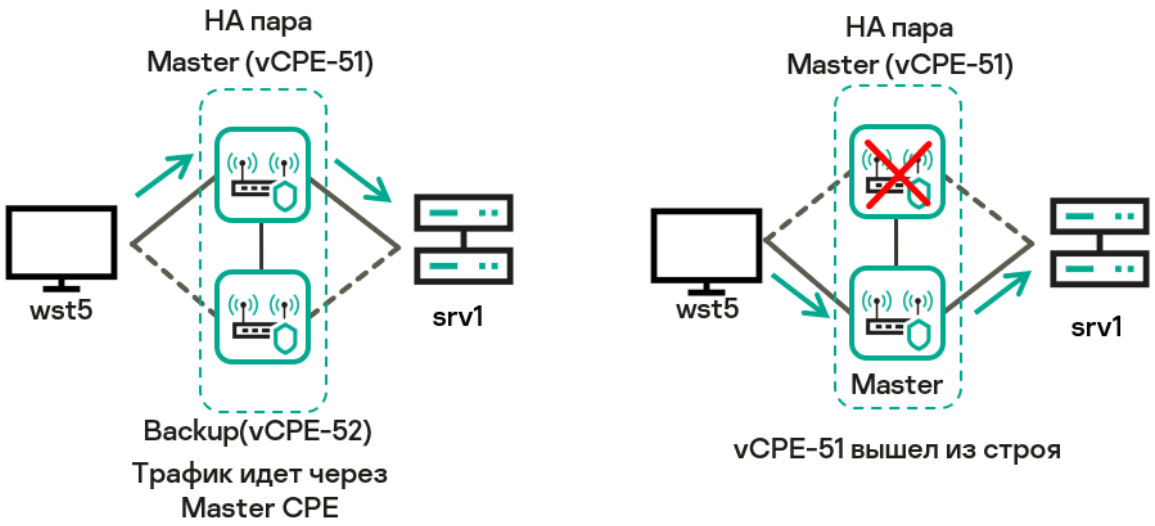


Рис. 5.2 Резервирование CPE с помощью протокола VRRP.

Взаимодействие по протоколу VRRP может быть настроено между несколькими устройствами CPE, а также между устройством CPE и сторонним маршрутизатором.

Для настройки VRRP необходимо создать экземпляры VRRP (VRRP instances), которые определяют, какие устройства CPE объединяются в виртуальные маршрутизаторы для обеспечения высокой доступности. При создании каждого экземпляра VRRP указываются общие параметры протокола VRRP, такие как идентификатор VRID (Virtual Router Identifier) виртуального маршрутизатора и виртуальный IP-адрес для сетевого интерфейса устройства CPE.

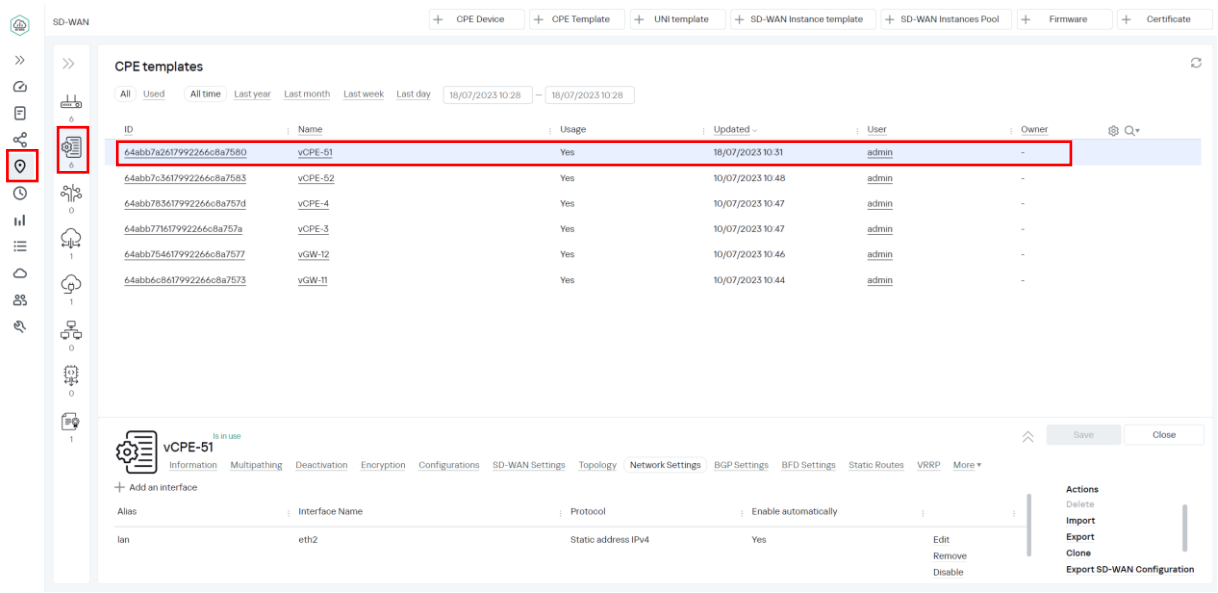
Экземпляры VRRP могут быть объединены в группы для синхронизации их работы. Таким образом, если в одном из экземпляров VRRP, входящих в группу, произойдет изменение основного VRRP-маршрутизатора, то основной маршрутизатор изменится во всех остальных экземплярах VRRP в группе.

Для получения дополнительной информации обратитесь к Kaspersky SD-WAN Online Help > Протокол VRRP: https://support.kaspersky.com/help/SD-WAN/2.1/ru-RU/246585.htm

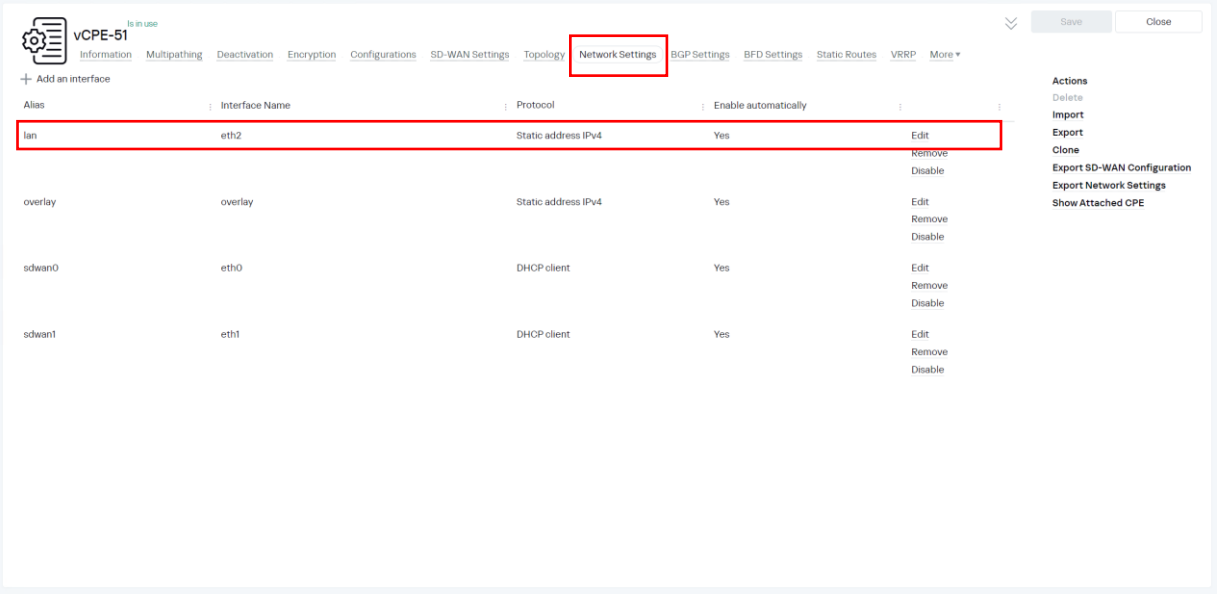
В данном сценарии будет настроен экземпляр VRRP между vCPE-51 и vCPE-52 и проверена работа протокола путем отключения lan интерфейса(eth2) между vCPE-51 и wst5.

5.2.1. Настройка адреса lan интерфейса.

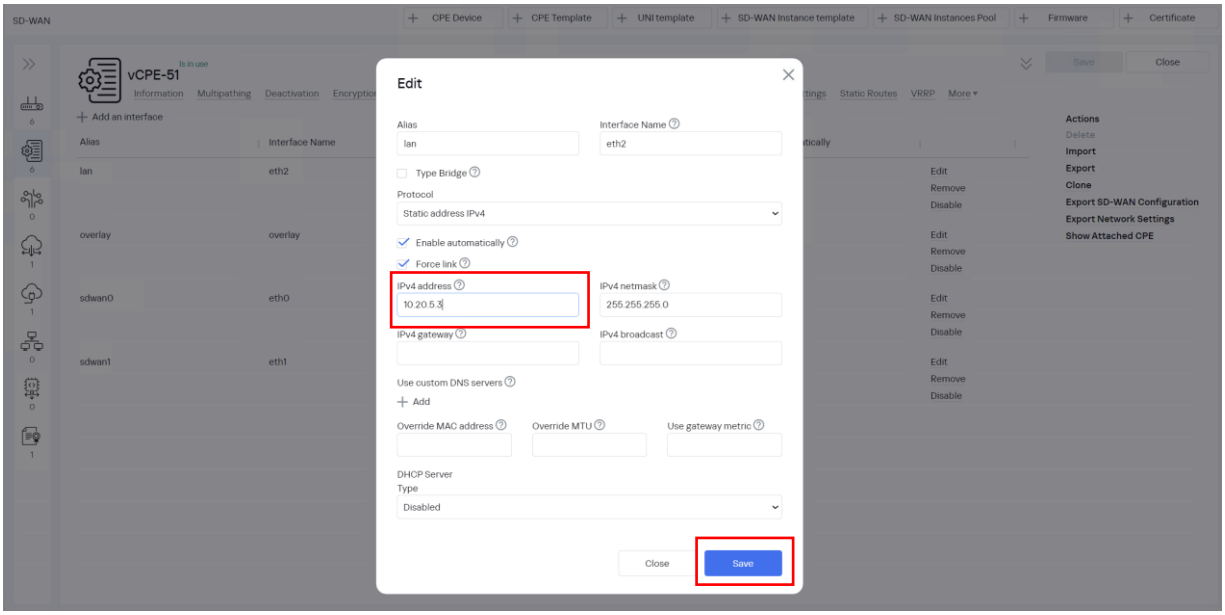
Для выполнения сценария необходимо изменить IP-адрес lan интерфейса на устройстве vCPE-51.
Перейти в меню SD-WAN, открыть CPE templates и выбрать vCPE-51



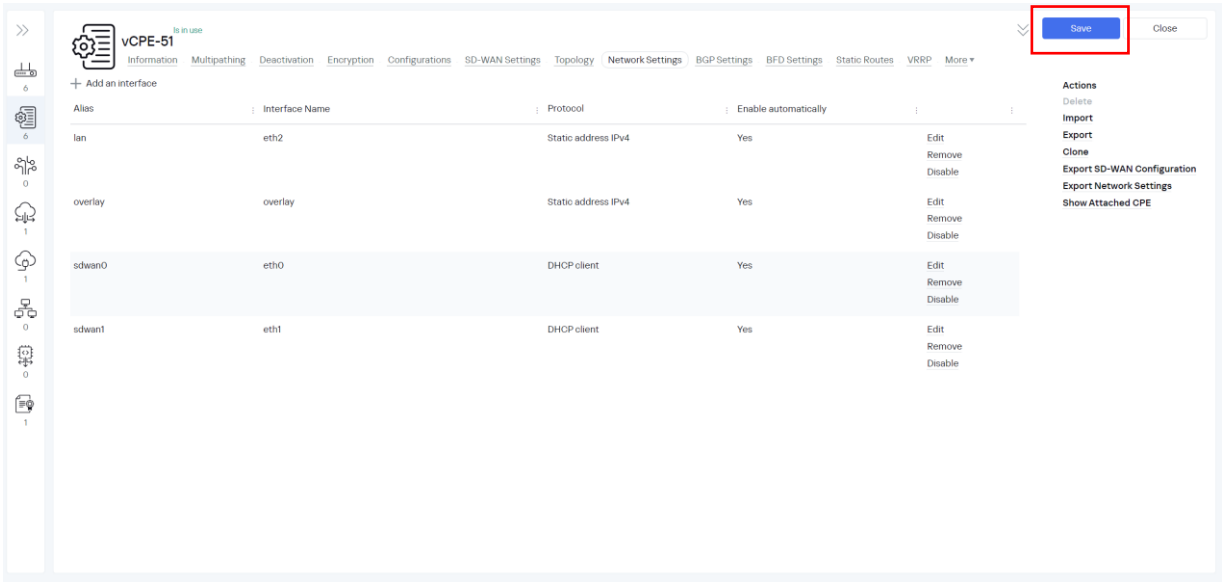
Открыть вкладку Network settings. Для интерфейса lan выбрать Edit.



Изменить адрес на 10.20.5.3 в меню настройки интерфейса. Нажать Save.

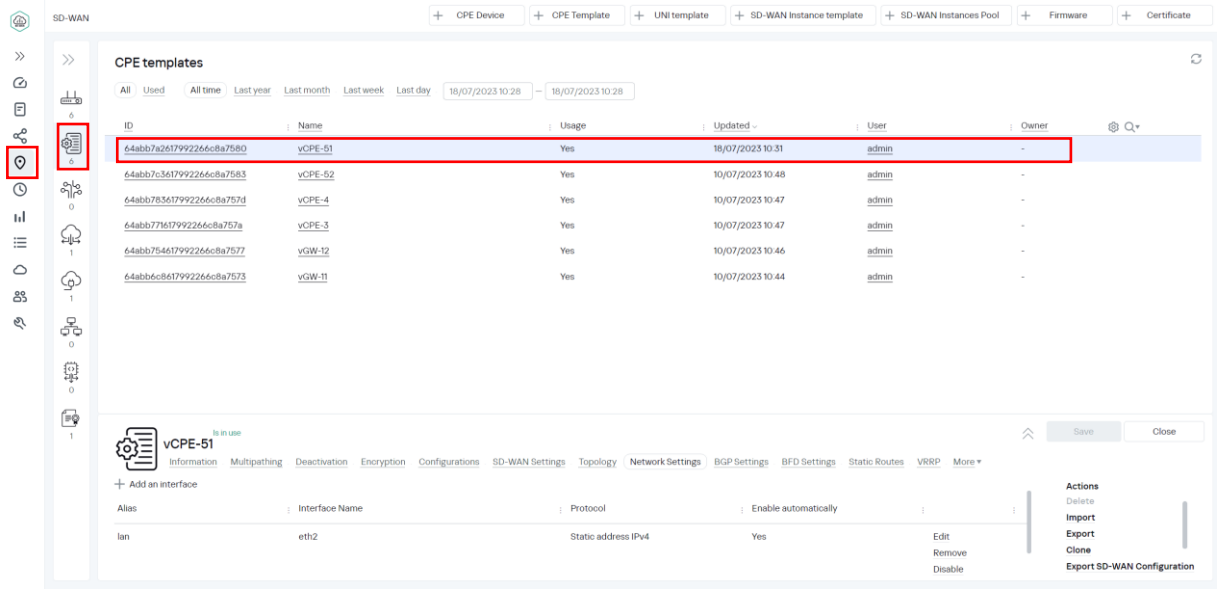


Нажать Save в шаблоне CPE.

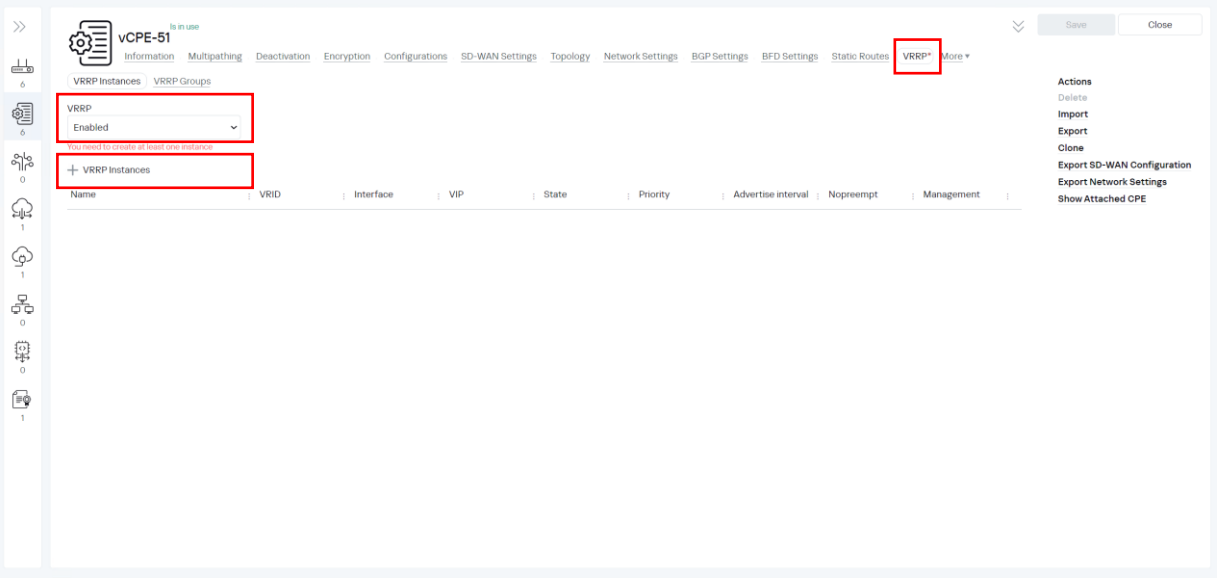


5.2.2. Настройка экземпляра VRRP.

Перейти в меню SD-WAN, открыть CPE templates и выбрать vCPE-51



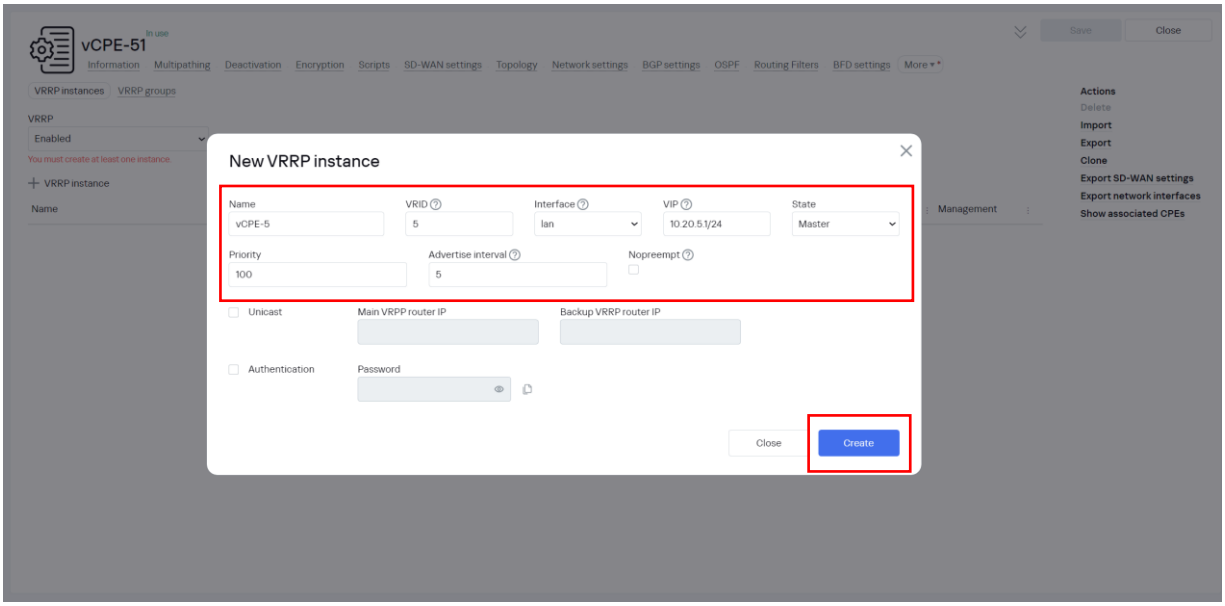
Открыть вкладку VRRP. Установить у параметра VRRP значение Enabled и нажать на +VRRP Instances для создания экземпляра VRRP.



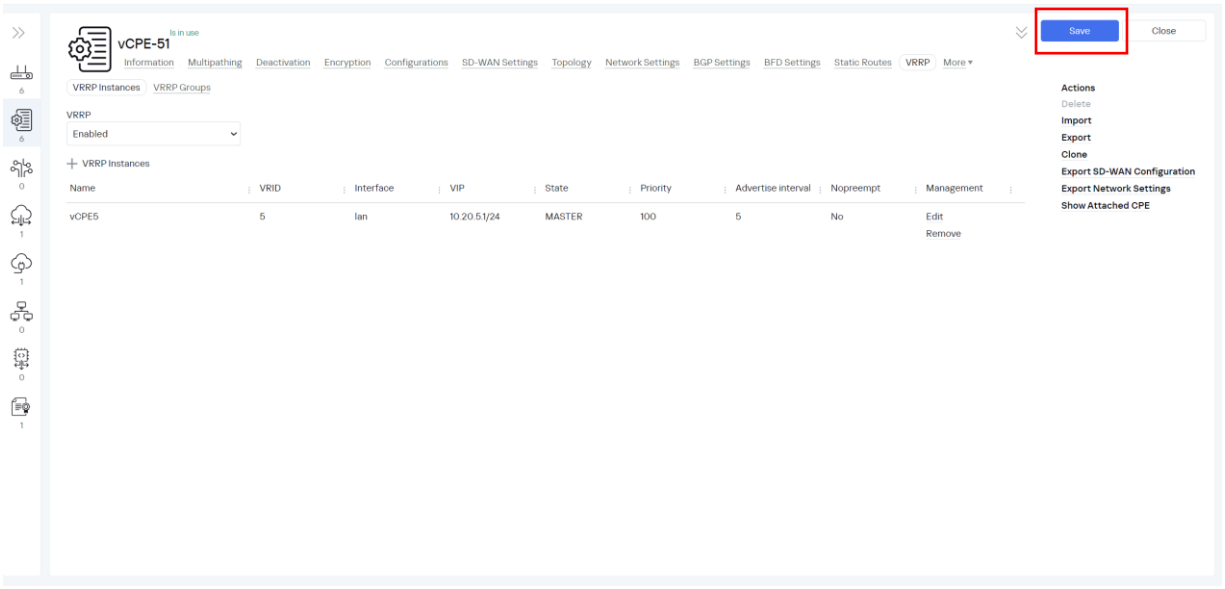
Задать параметры для экземпляра VRRP:

- Имя – vCPE5
- VRID – 5 (идентификатор экземпляра, должен совпадать у всех устройств VRRP группы).
- Interface – lan (сетевой интерфейс, который ассоциируется с VRRP).
- VIP – 10.20.5.1/24 (виртуальный IP адрес, который будет назначен основному маршрутизатору группы).
- State – MASTER (состояние маршрутизатора, в данном сценарии vCPE-51 будет основным).
- Priority – 100 (приоритет маршрутизатора, более высокий приоритет означает, что маршрутизатор будет выбран основным).

Нажать Create.



Нажать Save в шаблоне CPE.

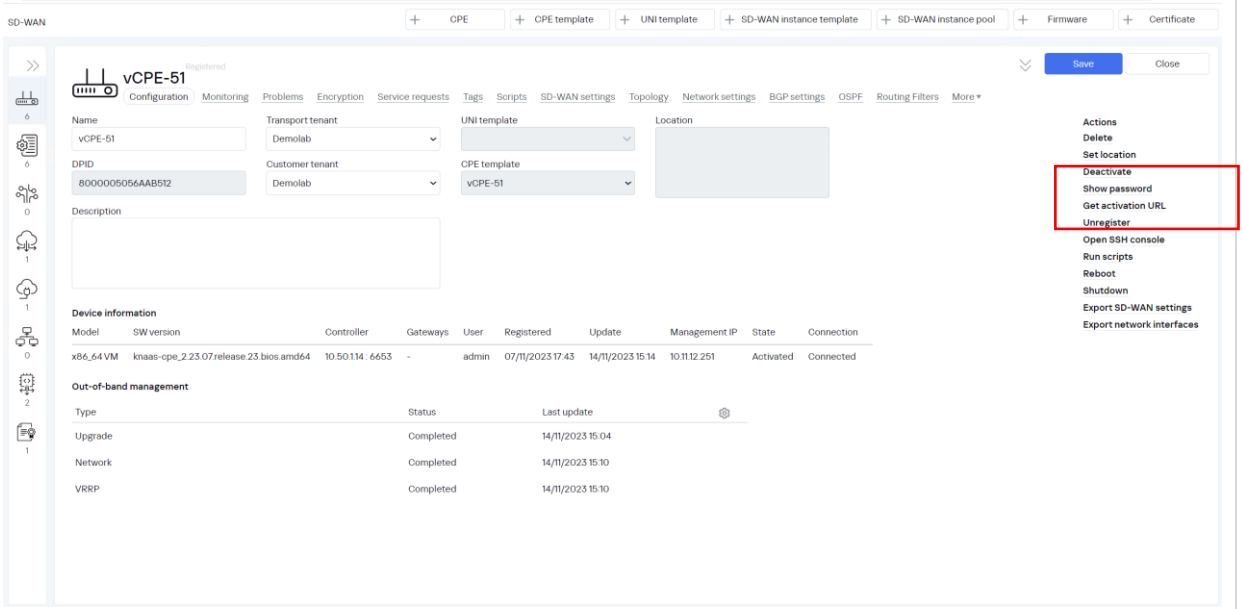


5.2.3. Повторить пункт 5.2.2 для vCPE-52 со следующими параметрами VRRP:

- Имя – vCPE5
- VRID – 5 (идентификатор экземпляра, должен совпадать у всех устройств VRRP группы).
- Interface – lan (сетевой интерфейс, который ассоциируется с VRRP).
- VIP – 10.20.5.1/24 (виртуальный IP адрес, который будет назначен основному маршрутизатору группы).
- State – BACKUP (состояние маршрутизатора, в данном сценарии vCPE-52 является резервным).
- Priority – 50 (приоритет маршрутизатора, более высокий приоритет означает, что маршрутизатор будет выбран основным).

5.2.4. Проверка работы экземпляра VRRP.

Подключится к vCPE-51. Для просмотра пароля от CPE выбрать vCPE-51 и нажать Show password.



Проверить, что маршрутизатор назначил виртуальный IP-адрес на интерфейс lan.

 # ip a | grep eth2 -A 3

```
root@8000005056AAB512:~# ip a | grep eth2 -A 3
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:50:56:aa:dd:49 brd ff:ff:ff:ff:ff:ff
    inet 10.20.5.3/24 brd 10.20.5.255 scope global eth2
       valid_lft forever preferred_lft forever
    inet 10.20.5.1/24 scope global secondary eth2
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:feaa:dd49/64 scope link
       valid_lft forever preferred_lft forever
root@8000005056AAB512:~#
```

Подключится к хосту wst5 и проверить связность с wst3:

[root@wst5 ~]$ ping 10.20.3.11

```
[ivpanin@wst5 ~]$ ping 10.20.3.11
PING 10.20.3.11 (10.20.3.11) 56(84) bytes of data.
64 bytes from 10.20.3.11: icmp_seq=1 ttl=62 time=3.63 ms
64 bytes from 10.20.3.11: icmp_seq=2 ttl=62 time=3.62 ms
64 bytes from 10.20.3.11: icmp_seq=3 ttl=62 time=2.93 ms
64 bytes from 10.20.3.11: icmp_seq=4 ttl=62 time=3.32 ms
64 bytes from 10.20.3.11: icmp_seq=5 ttl=62 time=3.17 ms
64 bytes from 10.20.3.11: icmp_seq=6 ttl=62 time=2.17 ms
64 bytes from 10.20.3.11: icmp_seq=7 ttl=62 time=2.09 ms
64 bytes from 10.20.3.11: icmp_seq=8 ttl=62 time=3.34 ms
64 bytes from 10.20.3.11: icmp_seq=9 ttl=62 time=3.40 ms
64 bytes from 10.20.3.11: icmp_seq=10 ttl=62 time=3.24 ms
64 bytes from 10.20.3.11: icmp_seq=11 ttl=62 time=3.57 ms
64 bytes from 10.20.3.11: icmp_seq=12 ttl=62 time=3.48 ms
64 bytes from 10.20.3.11: icmp_seq=13 ttl=62 time=3.79 ms
64 bytes from 10.20.3.11: icmp_seq=14 ttl=62 time=3.58 ms
```

Отключить lan интерфейс (eth2) на vCPE-51:

 # ip link set dev eth2 down

Подключится к vCPE-52. Для просмотра пароля от CPE выбрать vCPE-52 и нажать Show password.

Проверить, что данный маршрутизатор назначил виртуальный IP-адрес на lan интерфейс (eth2).

 # ip a | grep eth2 -A 3

```
root@8000005056AAC6B5:~# ip a | grep eth2 -A 3
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group defa
ult qlen 1000
    link/ether 00:50:56:aa:f1:04 brd ff:ff:ff:ff:ff:ff
    inet 10.20.5.2/24 brd 10.20.5.255 scope global eth2
       valid_lft forever preferred_lft forever
    inet 10.20.5.1/24 scope global secondary eth2
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:feaa:f104/64 scope link
       valid_lft forever preferred_lft forever
root@8000005056AAC6B5:~#
```

Подключится к хосту wst5 и проверить связность с wst3:

[root@wst5 ~]$ ping 10.20.3.11

```
[ivpanin@wst5 ~]$ ping 10.20.3.11
PING 10.20.3.11 (10.20.3.11) 56(84) bytes of data.
64 bytes from 10.20.3.11: icmp_seq=1 ttl=62 time=2.73 ms
64 bytes from 10.20.3.11: icmp_seq=2 ttl=62 time=3.15 ms
64 bytes from 10.20.3.11: icmp_seq=3 ttl=62 time=2.91 ms
64 bytes from 10.20.3.11: icmp_seq=4 ttl=62 time=2.68 ms
64 bytes from 10.20.3.11: icmp_seq=5 ttl=62 time=2.34 ms
64 bytes from 10.20.3.11: icmp_seq=6 ttl=62 time=3.08 ms
64 bytes from 10.20.3.11: icmp_seq=7 ttl=62 time=3.15 ms
64 bytes from 10.20.3.11: icmp_seq=8 ttl=62 time=2.95 ms
64 bytes from 10.20.3.11: icmp_seq=9 ttl=62 time=3.48 ms
64 bytes from 10.20.3.11: icmp_seq=10 ttl=62 time=3.35 ms
64 bytes from 10.20.3.11: icmp_seq=11 ttl=62 time=3.23 ms
64 bytes from 10.20.3.11: icmp_seq=12 ttl=62 time=3.36 ms
64 bytes from 10.20.3.11: icmp_seq=13 ttl=62 time=3.28 ms
64 bytes from 10.20.3.11: icmp_seq=14 ttl=62 time=3.10 ms
64 bytes from 10.20.3.11: icmp_seq=15 ttl=62 time=3.26 ms
```

Как видно, связность между рабочими станциями сохранилась и VRRP отработал корректно.

5.2.5. Возврат настроек после завершения теста.

Требуется включить интерфейс lan(eth2) на vCPE-51.

Подключиться к vCPE-51 и выполнить:

 # ip link set dev eth2 up

Проверить, что данный маршрутизатор назначил виртуальный IP-адрес на lan интерфейс (eth2).

 # ip a | grep eth2 -A 3

```
root@8000005056AAB512:~# ip a | grep eth2 -A 3
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:50:56:aa:dd:49 brd ff:ff:ff:ff:ff:ff
    inet 10.20.5.3/24 brd 10.20.5.255 scope global eth2
       valid_lft forever preferred_lft forever
    inet 10.20.5.1/24 scope global secondary eth2
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:feaa:dd49/64 scope link
       valid_lft forever preferred_lft forever
root@8000005056AAB512:~#
```

kaspersky

Приложение А.

Checklist.

Перед выполнением тестов должны быть выполнены все настройки из документа Proof of Concept Руководство по настройке демонстрационного стенда Kaspersky SD-WAN 2.0 Часть 1.

N	Название теста	Пункт настройки	Ожидаемый результат	Результат проверки (пройден/не пройден)
1	**Управление трафиком.**			
1.1	Балансировка нагрузки в режиме Active / Active.	3.1	Трафик балансируется между двумя WAN интерфейсами устройства vCPE-3.	
1.2	Резервирование каналов связи в режиме Active/Standby.	3.2	При работающем основном WAN интерфейсе устройства vCPE-3 трафик не идет через резервный WAN интерфейс. При отключении основного WAN-интерфейса на устройстве vCPE-3 трафик переключается на резервный WAN-интерфейс.	
1.3	Резервирование каналов связи в широковещательном (broadcast) режиме.	3.3	Копии пакетов с устройства vCPE-3 отправляются по интерфейсам genev_sys_4800/4801 в сторону vGW-11/12.	
1.4	Использование механизма FEC.	3.4	При включении FEC уменьшается процент потерь пакетов на интерфейсе, для которого включена эмуляция потерь.	
1.5	Включение мониторинга потерь пакетов на туннелях.	3.4.2-3.4.4	При включении мониторинга потерь в оркестраторе отображается статистика потерь для туннелей.	
1.6	Включение мониторинга задержек и джиттера на туннелях.	3.5.2-3.5.7	При включении мониторинга задержек и джиттера в оркестраторе отображается статистика задержек и джиттера для туннелей.	
1.7	Управление трафиком с помощью ограничений (Constraints).	3.5	При применении ограничений на транспортный сервис из пути прохождения трафика исключаются туннели, не удовлетворяющие заданным условиям (задаются пороговые значения задержки и джиттера). В статистике iperf уменьшается значения джиттера для	

			трафика, проходящего от устройства vCPE-3 к vCPE-4.	
1.8	Классификация трафика с помощью ACL и перенаправления в туннели, соответствующих заданным ограничениями.	3.6	Трафик, подпадающий под параметры созданного ACL (protocol UDP, port 5555), перенаправляется в туннели, не отмеченные как “Unsolicited”.	
1.9	Классификация трафика с помощью DPI и перенаправления в туннели, соответствующие заданным ограничениями.	3.7	Трафик, подпадающий под параметры созданного DPI ACL (SSH и HTTP), перенаправляется в туннели, не отмеченные как “ Unsolicited ”.	

2	**Построение топологии SD-WAN сети.**			
2.1	Создание топологий Full-Mesh.	4.1	После настройки топологических тегов, устройства CPE создают дополнительные туннели для построения Full-Mesh топологии (от каждого устройства CPE созданы туннели до всех других устройств CPE).	
2.2	Создание топологий Partial-Mesh.	4.2	После настройки топологических тегов, устройства CPE создают дополнительные туннели для построения Partial-Mesh топологии. Созданы 2 группы CPE: vCPE-3 и vCPE-4, и vCPE-51, vCPE-52, vCPE-4. CPE данных группы строят прямые туннели до всех устройств в своей группе.	
2.3	Создание топологий с использованием транзитных CPE.	4.3	Устройства vCPE-3 и vCPE-51 строят туннели через устройство vCPE-4, отмеченное как транзитное.	

3	**Работа с CPE устройствами.**			
3.1	Централизованное обновление firmware CPE устройств.	5.1	Firmware успешно загружено в оркестратор. Для CPE отображается доступность новой версии firmware. Обновление CPE на новую версию firmware проходит успешно: CPE после загрузки успешно подключаются к контроллеру и в интерфейсе оркестратора отображается новая версия firmware.	
3.2	Резервирование устройств CPE с использованием VRRP.	5.2	На паре устройств vCPE-51/52 успешно применяются настройки VRRP. На VRRP Master (vCPE-51) появляется настроенный виртуальный IP и обеспечивается связность между хостами wst5 и srv1. При отключении lan интерфейса на vCPE-51, виртуальный IP адрес переходит на устройство vCPE-52, которое становится VRRP Master, при этом также обеспечивается связность между хостами wst5 и srv1.	

https://kaspersky.ru/
https://securelist.ru