



## Руководство по дизайну IP-фабрики на коммутаторах Eltex

1	История версий .....	4
2	Цель.....	6
3	Используемые термины и сокращения .....	7
4	Архитектура Spine-Leaf.....	9
5	Предварительная настройка.....	11
6	Настройка underlay с использованием протокола OSPF .....	14
6.1	Настройка Spine .....	15
6.2	Настройка Leaf .....	17
6.3	Проверка настроек underlay .....	19
6.4	Ожидаемый результат .....	20
7	Настройка underlay с использованием протокола IS-IS.....	21
7.1	Настройка Spine .....	21
7.2	Настройка Leaf .....	23
7.3	Проверка настроек underlay .....	24
7.4	Ожидаемый результат .....	26
8	Балансировка трафика.....	27
8.1	Общие положения .....	27
8.2	Параметры hash-функции .....	27
9	Настройка overlay. VXLAN .....	28
9.1	Настройка VXLAN.....	28
9.2	Проверка настройки VXLAN .....	29
9.3	ARP suppression .....	34
9.4	Ожидаемый результат .....	35
10	Работа EVPN/VXLAN поверх eBGP .....	36
11	Настройка overlay. Multicast VXLAN .....	37
11.1	Настройка Spine .....	37
11.2	Настройка Leaf .....	38
11.3	Настройка multicast vxlan .....	39
11.4	Проверка настройки multicast vxlan.....	42
11.5	Ожидаемый результат .....	44
12	Настройка overlay. Symmetric IRB .....	45
12.1	Настройка symmetric IRB .....	45
12.2	Проверка настройки .....	48

12.3	Anycast gateway .....	50
12.4	Gateway-ip для маршрутов типа 5.....	52
12.5	DHCP relay .....	53
12.6	Ожидаемый результат .....	58
13	EVPN multihoming.....	59
13.1	Настройка EVPN multihoming.....	60
13.2	Core isolation.....	61
13.3	Проверка настройки .....	62
13.4	Ожидаемый результат .....	64
14	Команды для просмотра информации .....	65
15	Сеть управления .....	74
15.1	Схема сети OOB .....	74
15.2	Конфигурации устройств .....	76
16	Приложение 1 .....	81
16.1	Конфигурации с использованием протокола OSPF.....	81
16.2	Конфигурации с использованием протокола IS-IS .....	91
16.3	Конфигурации для multicast VXLAN .....	101
16.4	Конфигурации для Symmetric IRB .....	114
16.5	Конфигурации для EVPN multihoming .....	132

## 1 История версий

Версия документа	Дата выпуска	Содержание изменений
Версия 1.9	08.05.2026	Добавлены разделы: 5. Предварительная настройка 8. Балансировка трафика 10. Работа EVPN/VXLAN поверх eBGP 14. Команды для просмотра информации Изменения в разделах: 6. Настройка underlay с использованием протокола OSPF 7. Настройка underlay с использованием протокола IS-IS 13. EVPN multihoming Приложение 1
Версия 1.8	11.04.2025	Изменения в разделах: 9. Настройка overlay. Symmetric IRB 10. EVPN multihoming Приложение 1
Версия 1.7	25.12.2024	Изменения в разделах: 6. Настройка underlay с использованием протокола OSPF 10. EVPN multihoming Приложение 1
Версия 1.6	01.11.2024	Изменения в разделах: 7. Настройка overlay. VXLAN 9. Настройка overlay. Symmetric IRB
Версия 1.5	12.08.2024	Изменения в разделах: 9. Настройка overlay. Symmetric IRB Приложение 1
Версия 1.4	27.04.2024	Изменения в разделах: 9. Настройка overlay. Symmetric IRB Приложение 1
Версия 1.3	21.12.2023	Изменения в разделах: 9. Настройка overlay. Symmetric IRB

<b>Версия документа</b>	<b>Дата выпуска</b>	<b>Содержание изменений</b>
Версия 1.2	28.07.2023	Изменения в разделах: 2. Используемые термины и сокращения 7.1 Настройка VXLAN 7.2 Проверка настройки VXLAN Добавлены разделы: 8. Настройка overlay. Multicast VXLAN 9. Настройка overlay. Symmetric IRB 10. EVPN multihoming
Версия 1.1	30.09.2022	Первая публикация

## 2 Цель

Цель данного руководства – предоставить читателю основные инструменты, необходимые для построения IP-фабрики на базе оборудования Eltex. В основе данной IP-фабрики используется технология EVPN/VXLAN.

Целевой аудиторией являются сетевые специалисты, системные интеграторы, партнеры и заказчики, использующие или планирующие использовать оборудование производства Eltex.

### 3 Используемые термины и сокращения

**Anycast gateway** — механизм адресации шлюза по умолчанию, который позволяет использовать одни и те же IP- и MAC-адреса шлюза на всех устройствах Leaf, являющихся членами одного и того же L3VNI.

**ARP suppression** — подавление ARP. Данная функция позволяет устройству Leaf отвечать на ARP-запрос от имени удаленного хоста, не пересылая ARP-запрос через VXLAN.

**BFD (Bidirectional Forwarding Detection protocol)** — протокол, созданный для быстрого обнаружения неисправностей линков.

**ECMP (Equal-cost multi-path routing)** — технология балансировки нагрузки, позволяющая передавать пакеты одному получателю по нескольким «лучшим маршрутам». Данный функционал предназначен для распределения нагрузки и оптимизации пропускной способности сети.

**ESI** — ethernet segment identifier — уникальный в пределах IP-фабрики идентификатор Ethernet-сегмента. Имеет длину 10 байт.

**Ethernet-segment (ES)** — представляет собой совокупность линков, образующих агрегированный канал (LAG), соединяющий клиентское устройство с группой Leaf.

**EVPN (Ethernet Virtual Private Network)** — стандарт, определенный в RFC 7432. EVPN — это расширение протокола BGP (address-family, AFI: 25, SAFI: 70), функционирующее как плоскость управления для создания L2VPN- и L3VPN-сервисов. Расширение позволяет сети передавать информацию о конечных устройствах, такую как MAC- и IP-адреса.

**IBGP (Internal BGP)** — используется для соединения BGP-соседей в пределах одной автономной системы.

**IGP (Interior Gateway Protocol)** — протокол внутреннего шлюза (например IS-IS, OSPF). IGP-протоколы используются для передачи информации о маршрутах в пределах автономной системы.

**Ingress replication** — режим работы VXLAN, при котором репликация BUM-трафика осуществляется на входящем VTEP.

**IP-фабрика** — сетевая инфраструктура, основанная на протоколе IP и позволяющая создавать несколько симметричных путей между всеми устройствами в IP-фабрике.

**L3VNI** — VNI, используемый для маршрутизации.

**Leaf** — устройство уровня доступа в IP-фабрике.

**MSDP** — протокол обнаружения источников многоадресной рассылки. В рамках IP-фабрики используется для обмена информацией об источниках multicast-трафика между Spine.

**Multicast VXLAN** — режим работы VXLAN, при котором репликация BUM-трафика осуществляется посредством PIM multicast.

**OOB-интерфейс** — отдельный порт на устройстве для удаленного управления. Управление осуществляется по сети отдельно с каналом передачи данных.

**Overlay-сеть** — логическая сеть, созданная поверх другой (underlay) сети и использующая её инфраструктуру как транспорт.

**PIM** — протокол многоадресной маршрутизации для IP-сетей.

**POD (Point Of Delivery)** — обособленная группа устройств в топологии Клоза (Spine первого уровня и Leaf), Spine которой имеют связи только с Leaf своей группы и не имеют связи с Leaf остальной IP-фабрики.

**Route target (RT)** — расширенное BGP community.

**Spine** — центральное устройство в IP-фабрике, имеющее подключения ко всем Leaf (к Leaf своего POD в случае наличия POD в схеме).

**Underlay сеть** – базовая физическая сеть, обеспечивающая возможность соединения между всеми устройствами.

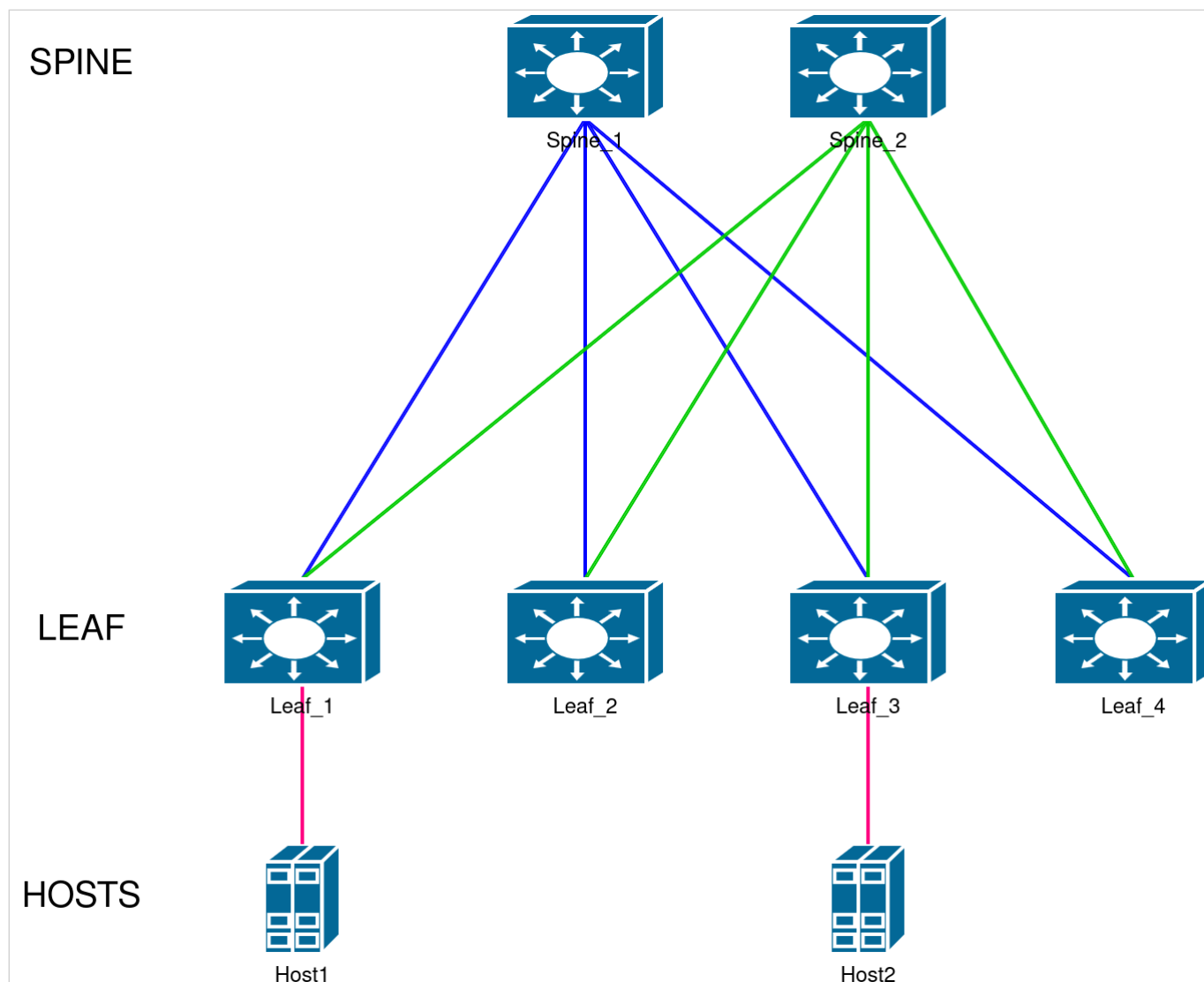
**VNI (Virtual Network Identifier)** – идентификатор сети в рамках VXLAN. Имеет длину 24 бита.

**VTEP (Virtual Tunnel End Point)** – устройство, на котором начинается или заканчивается VXLAN-туннель. Трафик инкапсулируется в VXLAN на VTEP-источнике, а декапсуляция выполняется на удаленном VTEP.

**VXLAN (Virtual eXtensible Local Area Network)** – виртуальная расширенная частная сеть. Описана в стандарте RFC 7348. Это технология создания виртуальной (наложенной) сети поверх существующей IP-инфраструктуры.

## 4 Архитектура Spine-Leaf

Ниже представлена примерная схема IP-фабрики, построенной с использованием архитектуры Spine-Leaf.

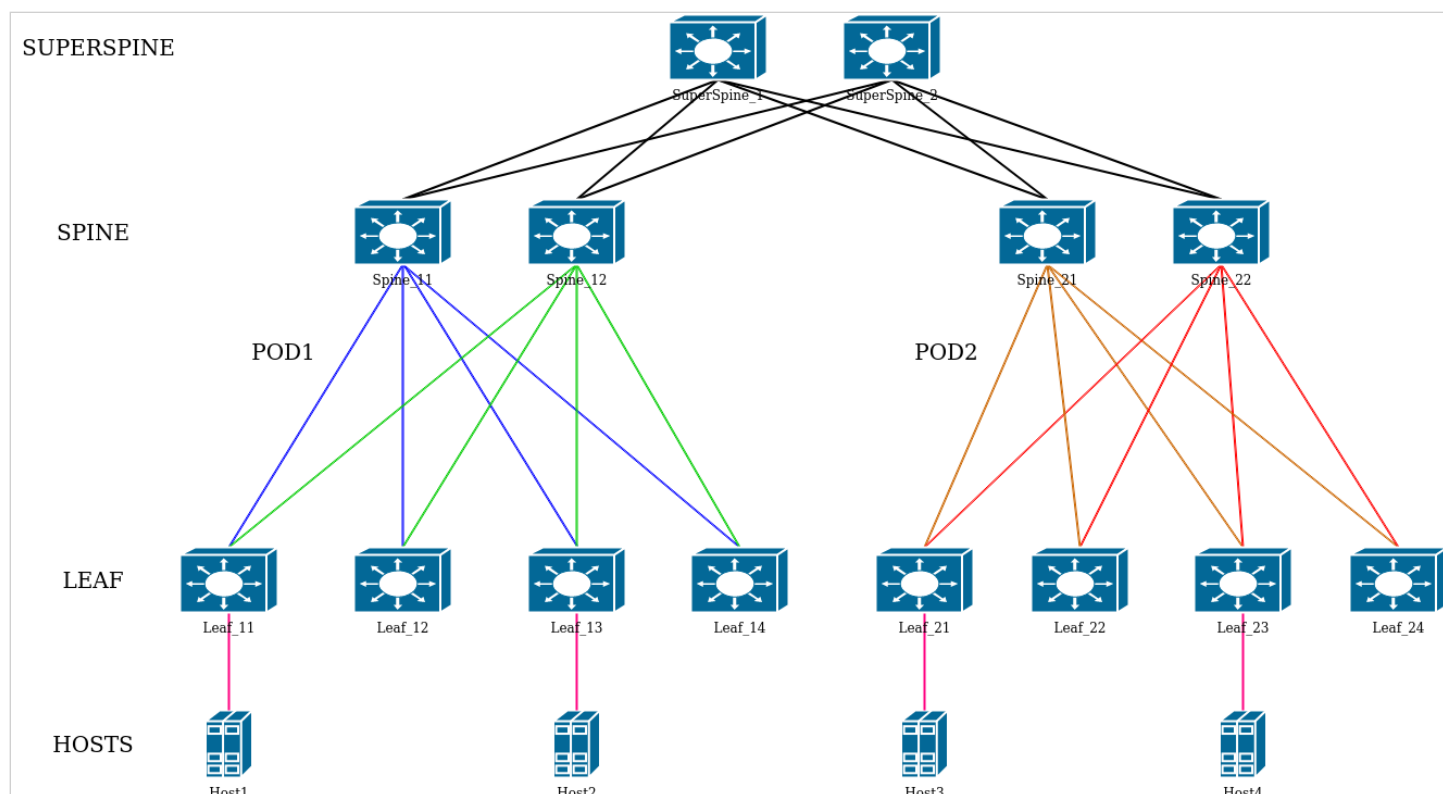


Второе название данной топологии – это *Folded Clos*, сложенная пополам сеть Клоза.

Устройства уровня Leaf в данной топологии выполняют роль коммутаторов доступа. К ним подключается оконечное оборудование. Leaf в свою очередь подключаются к устройствам уровня Spine (он же уровень коммутации).

Задача этого уровня – максимально быстрая маршрутизация пакетов с одного Leaf на другой. Таким образом между каждой парой Leaf существует количество равнозначных путей, равное количеству Spine.

В схеме ниже IP-фабрика, используемая для примера, увеличена в два раза.



Появилось разделение на POD (см. [Используемые термины и сокращения](#)) и дополнительный уровень коммутации — SuperSpine или Spine второго уровня. Данная IP-фабрика состоит из двух POD. На схеме видно, что POD обособлены друг от друга — у Spine первого уровня (на схеме это устройства Spine\_11-22) есть соединения только с определенной группой Leaf — Leaf своего POD. Связь между POD осуществляется через Spine второго уровня.

В схеме используются следующие протоколы и технологии:

- Ввиду использования в схеме EVPN, основным протоколом маршрутизации является BGP. Сессии BGP устанавливаются между loopback-интерфейсами устройств. Все устройства имеют единый номер автономной системы, образуя пространство iBGP.
- В настройках протокола BGP задействуется address-family L2VPN EVPN. Именно обмен сообщениями в рамках этой AF и обеспечивается работа технологии EVPN.
- В качестве IGP (протокола внутреннего шлюза) в данном руководстве используются протоколы IS-IS и OSPF. Основная задача IGP обеспечить IP-связность между всеми loopback-интерфейсами схемы, так как между ними строятся BGP-сессии и VXLAN-туннели.
- Для быстрого обнаружения неисправностей линков используется протокол BFD. Он позволяет определить неисправность линка менее чем за 1 секунду. Минимально возможное настраиваемое время реакции — 150 мс.
- BGP route reflector (RR). Настройка BGP, позволяющая маршрутизатору выступать в роли ретранслятора маршрутов. Так как в топологии Клоза отсутствует полная связность и используется iBGP, распространение маршрутной информации ограничено только прямыми линками. RR позволяет коммутаторам уровня Spine ретранслировать обновления маршрутной информации между коммутаторами уровня Leaf.
- Использование технологии ECMP в топологии Клоза обязательно. Благодаря ей неиспользуемые линки в схеме отсутствуют. Нагрузка равномерно распределяется по всем линкам, доступным между любой парой устройств Leaf. Повышается отказоустойчивость схемы.
- Jumbo-frame — поддержка передачи больших фреймов, до 10200 байт.
- Протоколы семейства Spanning Tree выключены на всех устройствах IP-фабрики.

## 5 Предварительная настройка

Некоторые настройки требуют перезагрузки устройства для применения. Чтобы свести число перезагрузок к одной, рекомендуется выполнить действия, описанные в данном разделе, до основной настройки устройств.

### Установка лицензий

Поддержка протокола BGP и технологии EVPN/VXLAN предоставляется по лицензии. Убедитесь в наличии соответствующих лицензий. Если лицензии отсутствуют, необходимо их установить.

Вывод команды **show license** в случае отсутствия установленных лицензий:

```
console#show license
Features installed:
      Feature          Licenses  Licenses
                   installed used       Active
-----
Licenses installed:
```


Для установки лицензии необходимо загрузить файл лицензии на устройство с помощью команды **boot license source\_url**. Пример:

```
console#boot license tftp://192.168.1.1/licensefile

console#boot license tftp://192.168.1.1/licensefile
02-Jun-2022 12:01:49 %COPY-I-FILECOPY: Files Copy - source URL tftp://192.168.1.1/licensefile
destination URL flash://system/licenses/licensefile
02-Jun-2022 12:01:49 %LICENSE-I-INSTALL: License file licensefile was installed
02-Jun-2022 12:01:49 %COPY-N-TRAP: The copy operation was completed successfully

Copy: 1181 bytes copied in 00:00:01 [hh:mm:ss]
```

### Настройки jumbo-frame и ECMP

-  Настройка jumbo-frame необходима для передачи через ip-фабрику пользовательских кадров большого размера, более 1500 байт. Максимальный размер MTU при этом составляет 10240 байт. Максимальный же размер пользовательского кадра для передачи через ip-фабрику составляет 10186 байт, т.к. для его передачи требуется еще 4 байта для контрольной суммы и 50 байт overhead при инкапсуляции в VXLAN.

Пример выполнения данных настроек в CLI:

```
console(config)#port jumbo-frame
This setting will take effect only after copying running configuration to startup configuration
and resetting the device
console(config)#ip maximum-paths 32
Warning! New value will be applied only after reboot
```

Где:

- port jumbo-frame – включение поддержки передачи больших фреймов;
- ip maximum-paths 32 – задание максимального количество путей, которые могут быть установлены в FIB для каждого маршрута, с помощью чего задействуется ECMP.

## Перезагрузка устройства

Для применения лицензии и настроек **port jumbo-frame** и **ip maximum-paths 32** необходимо сохранить конфигурацию и выполнить перезагрузку:

```
console#write
Overwrite file [startup-config].... (Y/N)[N] ?Y
23-Jun-2022 07:13:16 %COPY-I-FILECOPY: Files Copy - source URL running-config
destination URL flash://system/configuration/startup-config
23-Jun-2022 07:13:16 %COPY-N-TRAP: The copy operation was completed successfully
Copy succeeded
console#reload
This command will reset the whole system and disconnect your current session. Do
you want to continue ? (Y/N)[N] Y
Shutting down ...
```

После перезагрузки рекомендуется выполнить проверку успешности активации лицензии и применения настроек.

Вывод команды **show license** в случае успешно установленной лицензии:

```
console#show license
Features installed:

```

Feature	Licenses installed	Licenses used	Active
BGP	1	1	Yes
EVPN	1	1	Yes

```
-----
Licenses installed:
License name: licensefile
License version: 1.0
Valid for device: ES7A000000 (e0:d9:e3:00:00:00)
Status: Active
Features:
BGP, EVPN
```

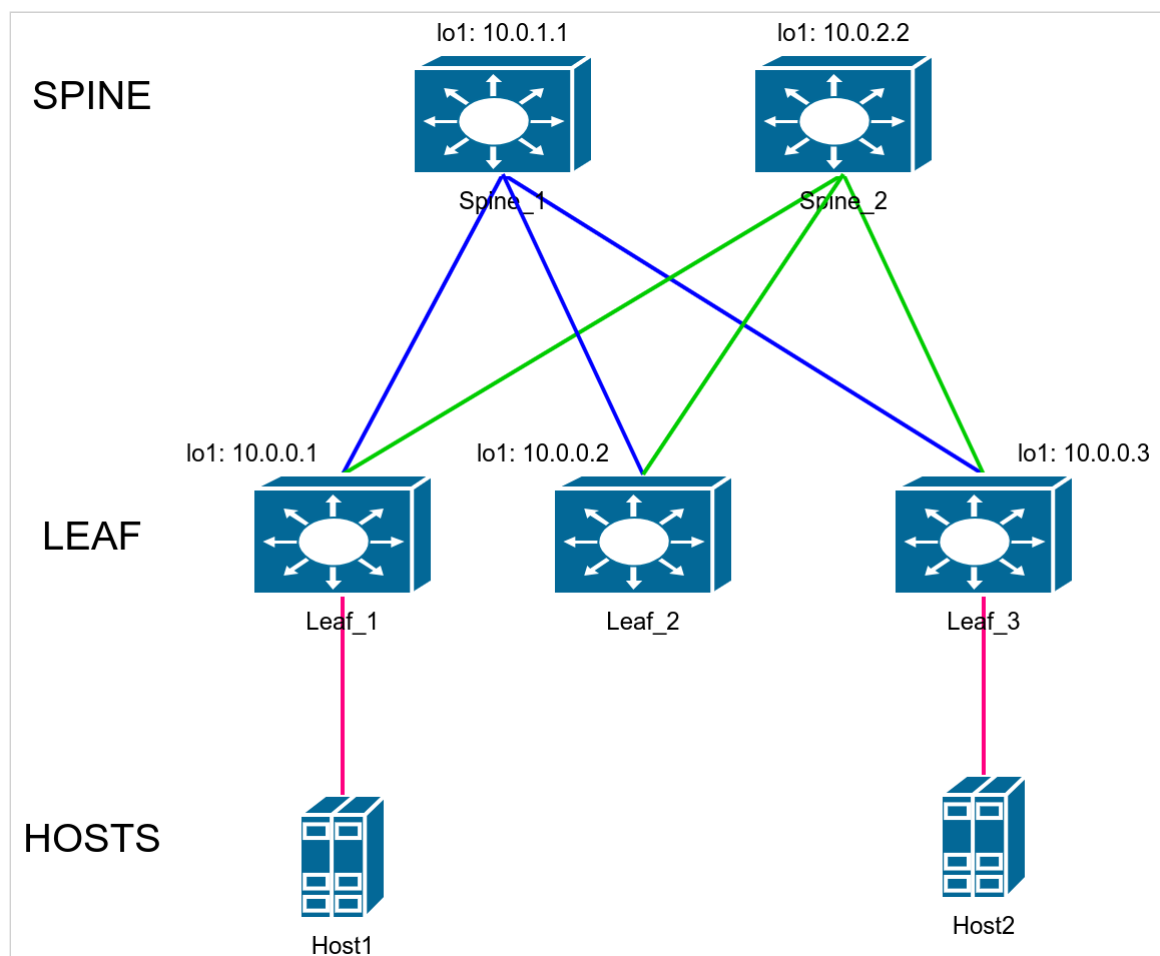
Для настроек **port jumbo-frame** и **ip maximum-paths**:

```
Spine_1#show ports jumbo-frame
Jumbo frames are enabled (current MTU 10240 bytes)
Jumbo frames will be enabled after reset (MTU 10240 bytes after reset)
Default protocol MTU will be set to 9000 after reset
Spine_1#show ip route
Maximum Parallel Paths: 32 (32 after reset)
Load balancing: src-dst-mac-ip
IP Forwarding: enabled
Codes: > - best, C - connected, S - static,
R - RIP,
O - OSPF intra-area, OIA - OSPF inter-area,
OE1 - OSPF external 1, OE2 - OSPF external 2,
B - BGP, i - IS-IS, L1 - IS-IS level-1,
L2 - IS-IS level-2, ia - IS-IS inter area
```

Строки **Jumbo frames are enabled (current MTU 10240 bytes)** и **Maximum Parallel Paths: 32 (32 after reset)** указывают на успешное включение соответствующих настроек.

## 6 Настройка underlay с использованием протокола OSPF

Ниже представлена схема для практической части.



⚠ IP-адресация в конфигурациях в данном руководстве используется для примера.

**Адреса loopback-интерфейсов (обозначены на схеме):**

Spine\_1: 10.0.1.1

Spine\_2: 10.0.2.2

Leaf\_1: 10.0.0.1

Leaf\_2: 10.0.0.2

Leaf\_3: 10.0.0.3

**Линковочные сети:**

Spine\_1-Leaf\_1: 172.16.1.0/30

Spine\_1-Leaf\_2: 172.16.2.0/30

Spine\_1-Leaf\_3: 172.16.3.0/30

Spine\_2-Leaf\_1: 172.16.1.4/30

Spine\_2-Leaf\_2: 172.16.2.4/30

Spine\_2-Leaf\_3: 172.16.3.4/30

## 6.1 Настройка Spine

Отключите протокол STP и задайте имя устройства.

```
console(config)#no spanning-tree
console(config)#hostname Spine_1
Spine_1(config)#
```

**!** Для упрощения процедуры настройки через консоль сначала можно использовать функцию **terminal no prompt**, отключающую необходимость подтверждения перед выполнением некоторых команд.

```
Spine_1#terminal no prompt
```

Настройка интерфейсов:

```
Spine_1(config)#interface HundredGigabitEthernet1/0/1
Spine_1(config-if)# description Leaf_1
Spine_1(config-if)# ip address 172.16.1.2 255.255.255.252
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface HundredGigabitEthernet1/0/2
Spine_1(config-if)# description Leaf_2
Spine_1(config-if)# ip address 172.16.2.2 255.255.255.252
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface HundredGigabitEthernet1/0/3
Spine_1(config-if)# description Leaf_3
Spine_1(config-if)# ip address 172.16.3.2 255.255.255.252
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface loopback1
Spine_1(config-if)# ip address 10.0.1.1 255.255.255.255
Spine_1(config-if)#exit
```

Задействование протокола маршрутизации OSPF:

**!** В целях минимизации перерыва в передаче трафика при изменении состояния линков Spine-Leaf, рекомендуется использовать настройку протокола OSPF **timers spf delay 0**. Она позволяет убрать задержку перед очередным расчетом SPF. Соответственно убирает задержку перед обновлением таблицы маршрутизации при изменениях в топологии OSPF.

```
Spine_1(config)#router ospf 1
Spine_1(router_ospf_process)# network 172.16.1.2 area 0.0.0.0
Spine_1(router_ospf_process)# network 172.16.2.2 area 0.0.0.0
Spine_1(router_ospf_process)# network 172.16.3.2 area 0.0.0.0
Spine_1(router_ospf_process)# router-id 10.0.1.1
Spine_1(router_ospf_process)# timers spf delay 0
Spine_1(router_ospf_process)# redistribute connected subnets
Spine_1(router_ospf_process)#exit
Spine_1(config)#interface ip 172.16.1.2
Spine_1(config-if)# ip ospf network point-to-point
Spine_1(config-if)#exit
Spine_1(config)#interface ip 172.16.2.2
Spine_1(config-if)# ip ospf network point-to-point
Spine_1(config-if)#exit
Spine_1(config)#interface ip 172.16.3.2
```

```
Spine_1(config-ip)# ip ospf network point-to-point
Spine_1(config-ip)#exit
```

⚠ В протоколе OSPF рекомендуем задействовать режим p2p (point-to-point). Он используется при установлении соединений на линковочных сетях, где каждый маршрутизатор имеет только одного соседа. На таком соединении нет необходимости выбирать Designated Router (DR) или Backup Designated Router (BDR). Этот подход ускоряет установление соседств, уменьшая время сходимости протокола.

Задействование протокола BGP:

⚠ Поддержка протокола BGP предоставляется по лицензии (см. [Установка лицензий](#)).

```
Spine_1(config)#router bgp 65500
Spine_1(router-bgp)# bgp router-id 10.0.1.1
This action will reset all neighbor connections and clear BGP routing table.
Spine_1(router-bgp)# address-family ipv4 unicast
Spine_1(router-bgp-af)# exit
Spine_1(router-bgp)# address-family l2vpn evpn
This action will reset all neighbor connections and clear BGP routing table.
Spine_1(router-bgp-af)# exit
Spine_1(router-bgp)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbrgrp)# remote-as 65500
This action will reset connection with all neighbors in peer group.
Spine_1(router-bgp-nbrgrp)# update-source loopback 1
This action will reset connection with all neighbors in peer group.
Spine_1(router-bgp-nbrgrp)# fall-over bfd
Spine_1(router-bgp-nbrgrp)# route-reflector-client
Spine_1(router-bgp-nbrgrp)# exit
Spine_1(router-bgp)# neighbor 10.0.0.1
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)# neighbor 10.0.0.2
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)# neighbor 10.0.0.3
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)#exit
```

⚠ Если в процессе настройки возникнет ситуация, когда BGP-соседство будет установлено раньше, чем включен route-reflector, то для применения настройки route-reflector-client необходим перезапуск процесса BGP:

```
Spine_1#clear ip bgp
This action will reset connection with the neighbor.
Would you like to continue? (Y/N)[N] Y
```

Настройка остальных устройств Spine в схеме выполняется аналогично, с внесением необходимых изменений согласно схемы и плану IP-адресации.

Изменению от устройства к устройству подлежат следующие параметры:

- Hostname устройства;
- Description на интерфейсах;
- IP-адреса интерфейсов (физических + loopback);
- IP-интерфейсы, на которых включен OSPF;
- OSPF router-id;
- BGP router-id;
- IP-адреса BGP-соседей.

## 6.2 Настройка Leaf

Отключите протокол STP и задайте имя устройства.

```
console(config)#no spanning-tree
console(config)#hostname Leaf_1
Leaf_1(config)#
```


Для упрощения процедуры настройки через консоль сначала можно использовать функцию **terminal no prompt**, отключающую необходимость подтверждения перед выполнением некоторых команд:

```
Leaf_1#terminal no prompt
```

Настройка интерфейсов:

```
Leaf_1(config)#interface HundredGigabitEthernet1/0/1
Leaf_1(config-if)# description Spine_1
Leaf_1(config-if)# ip address 172.16.1.1 255.255.255.252
This action will reset all neighbor connections on the interface.
Leaf_1(config-if)#exit
Leaf_1(config)#interface HundredGigabitEthernet1/0/2
Leaf_1(config-if)# description Spine_2
Leaf_1(config-if)# ip address 172.16.1.5 255.255.255.252
This action will reset all neighbor connections on the interface.
Leaf_1(config-if)#exit
Leaf_1(config)#interface loopback1
Leaf_1(config-if)# ip address 10.0.0.1 255.255.255.255
Leaf_1(config-if)#exit
```

Задействование протокола маршрутизации OSPF:

 В целях минимизации перерыва в передаче трафика при изменении состояния линков Spine-Leaf рекомендуется использовать настройку протокола OSPF **timers spf delay 0**. Она позволяет убрать задержку перед очередным расчетом SPF. Соответственно убирает задержку перед обновлением таблицы маршрутизации при изменениях в топологии OSPF.

```
Leaf_1(config)#router ospf 1
Leaf_1(router_ospf_process)#network 172.16.1.1 area 0.0.0.0
Leaf_1(router_ospf_process)#network 172.16.1.5 area 0.0.0.0
Leaf_1(router_ospf_process)#router-id 10.0.0.1
Leaf_1(router_ospf_process)#timers spf delay 0
```

```
Leaf_1(router_ospf_process)#redistribute connected subnets
Leaf_1(router_ospf_process)#exit
Leaf_1(config)#interface ip 172.16.1.1
Leaf_1(config-ip)#ip ospf network point-to-point
Leaf_1(config-ip)#exit
Leaf_1(config)#interface ip 172.16.1.5
Leaf_1(config-ip)#ip ospf network point-to-point
Leaf_1(config-ip)#exit
```

**⚠** В протоколе OSPF рекомендуем задействовать режим p2p (point-to-point). Он используется при установлении соединений на линковочных сетях, где каждый маршрутизатор имеет только одного соседа. На таком соединении нет необходимости выбирать Designated Router (DR) или Backup Designated Router (BDR). Этот подход ускоряет установление соседств, уменьшая время сходимости протокола.

Задействование протокола BGP:

**⚠** Поддержка протокола BGP предоставляется по лицензии (см. [Установка лицензий](#)).

```
Leaf_1(config)#router bgp 65500
Leaf_1(router-bgp)# bgp router-id 10.0.0.1
This action will reset all neighbor connections and clear BGP routing table.
Leaf_1(router-bgp)# address-family ipv4 unicast
Leaf_1(router-bgp-af)# exit
Leaf_1(router-bgp)# address-family l2vpn evpn
This action will reset all neighbor connections and clear BGP routing table.
Leaf_1(router-bgp-af)# exit
Leaf_1(router-bgp)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbrgrp)# remote-as 65500
This action will reset connection with all neighbors in peer group.
Leaf_1(router-bgp-nbrgrp)# update-source loopback 1
This action will reset connection with all neighbors in peer group.
Leaf_1(router-bgp-nbrgrp)# fall-over bfd
Leaf_1(router-bgp-nbrgrp)# exit
Leaf_1(router-bgp)# neighbor 10.0.1.1
Leaf_1(router-bgp-nbr)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbr)# address-family ipv4 unicast
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# exit
Leaf_1(router-bgp)# neighbor 10.0.2.2
Leaf_1(router-bgp-nbr)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbr)# address-family ipv4 unicast
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# exit
Leaf_1(router-bgp)#exit
```

Настройка остальных устройств Leaf в схеме выполняется аналогично, с внесением необходимых изменений согласно схемы и плану IP-адресации.

Изменению от устройства к устройству подлежат следующие параметры:

- Hostname устройства;
- Description на интерфейсах;
- IP-адреса интерфейсов (физических + loopback);
- IP-интерфейсы, на которых включен OSPF;
- OSPF router-id;
- BGP router-id;
- IP-адреса BGP-соседей.

### 6.3 Проверка настроек underlay

После выполнения вышеописанных настроек необходимо выполнить проверку установления соседств протоколов OSPF, BGP и BFD.

Используемые команды:

```
show ip ospf neighbor
show ip bgp neighbors
show ip bfd neighbors
```

Пример выполнения вышеуказанных show-команд на устройстве Spine\_1:

```
Spine_1#show ip ospf neighbor
Neighbor Addr      Neighbor ID      PID IP Interface  Pri State          Dead time Interface
-----
172.16.1.1        10.0.0.1        1 172.16.1.2     1 full/ -         00:00:36 hu1/0/1
172.16.2.1        10.0.0.2        1 172.16.2.2     1 full/ -         00:00:34 hu1/0/2
172.16.3.1        10.0.0.3        1 172.16.3.2     1 full/ -         00:00:37 hu1/0/3

Spine_1#show ip bgp neighbors

  BGP neighbor      Remote AS      Router ID      State          Uptime          Hold Time  Keepalive
-----
  10.0.0.1          65500         10.0.0.1      ESTABLISHED    00,00:09:03     90         30
  10.0.0.2          65500         10.0.0.2      ESTABLISHED    00,00:02:52     90         30
  10.0.0.3          65500         10.0.0.3      ESTABLISHED    00,00:01:45     90         30

Spine_1#show ip bfd neighbors

  Neighbor          Local          State          Last Down Diag
-----
  10.0.0.1          10.0.1.1      Up             No Diagnostic
  10.0.0.2          10.0.1.1      Up             No Diagnostic
  10.0.0.3          10.0.1.1      Up             No Diagnostic
```

Пример выполнения вышеуказанных show-команд на устройстве Leaf\_1:

```
Leaf_1#show ip ospf neighbor
Neighbor Addr      Neighbor ID      PID IP Interface  Pri State          Dead time Interface
-----
172.16.1.2        10.0.1.1        1 172.16.1.1     1 full/ -         00:00:37 hu1/0/1
172.16.1.6        10.0.2.2        1 172.16.1.5     1 full/ -         00:00:34 hu1/0/2

Leaf_1#show ip bgp neighbors
```

BGP neighbor	Remote AS	Router ID	State	Uptime	Hold Time	Keepalive
10.0.1.1	65500	10.0.1.1	ESTABLISHED	00,00:09:28	90	30
10.0.2.2	65500	10.0.2.2	ESTABLISHED	00,00:05:11	90	30

Leaf\_1#show ip bfd neighbors

Neighbor	Local	State	Last Down Diag
10.0.1.1	10.0.0.1	Up	No Diagnostic
10.0.2.2	10.0.0.1	Up	No Diagnostic

В случае успешного установления всех соседств в схеме должна быть обеспечена IP-связность между всеми устройствами. Проверить это можно, выполнив команду ICMP ping с любого устройства до любого другого, указав в качестве src и dst IP-адреса loopback-интерфейсов устройств.

Пример выполнения ping с Leaf\_1 до Leaf\_3:

```
Leaf_1#ping 10.0.0.3 source 10.0.0.1
Pinging 10.0.0.3 with 18 bytes of data:

18 bytes from 10.0.0.3: icmp_seq=1. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=2. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=3. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=4. time=0 ms

----10.0.0.3 PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms) min/avg/max = 0/0/0
```

## 6.4 Ожидаемый результат

- Все соседства протокола OSPF и BFD в состоянии UP. Соседства протокола BGP в состоянии ESTABLISHED;
- IP-связность между всеми устройствами схемы установлена.


## 7 Настройка underlay с использованием протокола IS-IS

Схема аналогична используемой в разделе [Настройка underlay с использованием протокола OSPF](#).

### 7.1 Настройка Spine

Отключите протокол STP и задайте имя устройства.


```
console(config)#no spanning-tree
console(config)#hostname Spine_1
Spine_1(config)#
```

 Для упрощения процедуры настройки через консоль сначала можно использовать функцию **terminal no prompt**, отключающую необходимость подтверждения перед выполнением некоторых команд.

```
Spine_1#terminal no prompt
```

Настройка интерфейсов:

```
Spine_1(config)#interface HundredGigabitEthernet1/0/1
Spine_1(config-if)# description Leaf_1
Spine_1(config-if)# ip address 172.16.1.2 255.255.255.252
Spine_1(config-if)# ip router isis
Spine_1(config-if)# isis network point-to-point
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface HundredGigabitEthernet1/0/2
Spine_1(config-if)# description Leaf_2
Spine_1(config-if)# ip address 172.16.2.2 255.255.255.252
Spine_1(config-if)# ip router isis
Spine_1(config-if)# isis network point-to-point
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface HundredGigabitEthernet1/0/3
Spine_1(config-if)# description Leaf_3
Spine_1(config-if)# ip address 172.16.3.2 255.255.255.252
Spine_1(config-if)# ip router isis
Spine_1(config-if)# isis network point-to-point
This action will reset all neighbor connections on the interface.
Spine_1(config-if)#exit
Spine_1(config)#interface loopback1
Spine_1(config-if)# ip address 10.0.1.1 255.255.255.255
Spine_1(config-if)#exit
```

 В протоколе IS-IS рекомендуем задействовать режим p2p (point-to-point). Он используется при установлении соединений на линковочных сетях, где каждый маршрутизатор имеет только одного соседа. Режим point-to-point уменьшает количество служебных пакетов протокола IS-IS, ускоряет установление соседств, уменьшая время сходимости протокола.

Задействование протокола маршрутизации IS-IS:

```
Spine_1(config)#router isis
Spine_1(router-isis)# address-family ipv4 unicast
Spine_1(router-isis-af)# redistribute connected
Spine_1(router-isis-af)# exit
Spine_1(router-isis)# net 49.0001.1111.1111.1111.00
```

```
Spine_1(router-isis)#exit
```

## Задействование протокола BGP:

**⚠** Поддержка протокола BGP предоставляется по лицензии (см. [Установка лицензий](#)).

```
Spine_1(config)#router bgp 65500
Spine_1(router-bgp)# bgp router-id 10.0.1.1
This action will reset all neighbor connections and clear BGP routing table.
Spine_1(router-bgp)# address-family ipv4 unicast
Spine_1(router-bgp-af)# exit
Spine_1(router-bgp)# address-family l2vpn evpn
This action will reset all neighbor connections and clear BGP routing table.
Spine_1(router-bgp-af)# exit
Spine_1(router-bgp)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbrgrp)# remote-as 65500
This action will reset connection with all neighbors in peer group.
Spine_1(router-bgp-nbrgrp)# update-source loopback 1
This action will reset connection with all neighbors in peer group.
Spine_1(router-bgp-nbrgrp)# fall-over bfd
Spine_1(router-bgp-nbrgrp)# route-reflector-client
Spine_1(router-bgp-nbrgrp)# exit
Spine_1(router-bgp)# neighbor 10.0.0.1
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)# neighbor 10.0.0.2
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)# neighbor 10.0.0.3
Spine_1(router-bgp-nbr)# peer-group LEAF_GROUP
Spine_1(router-bgp-nbr)# address-family ipv4 unicast
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Spine_1(router-bgp-nbr-af)# exit
Spine_1(router-bgp-nbr)# exit
Spine_1(router-bgp)#exit
```

**⚠** Если в процессе настройки возникнет ситуация, когда BGP-соседство будет установлено раньше, чем включен route-reflector, то для применения настройки route-reflector-client необходим перезапуск процесса BGP:

```
Spine_1#clear ip bgp
This action will reset connection with the neighbor.
Would you like to continue? (Y/N)[N] Y
```

Настройка остальных устройств Spine в схеме выполняется аналогично, с внесением необходимых изменений согласно схеме и плану IP-адресации.

Изменению от устройства к устройству подлежат следующие параметры:

- Hostname устройства;
- Description на интерфейсах;
- IP-адреса интерфейсов (физических + loopback);
- IS-IS net идентификатор;
- BGP router-id;
- IP-адреса BGP-соседей.

## 7.2 Настройка Leaf

Отключите протокол STP и задайте имя устройства.

```
console(config)#no spanning-tree
console(config)#hostname Leaf_1
Leaf_1(config)#
```

Для упрощения процедуры настройки через консоль сначала можно использовать функцию **terminal no prompt**, отключающую необходимость подтверждения перед выполнением некоторых команд:

```
Leaf_1#terminal no prompt
```

Настройка интерфейсов:

```
Leaf_1(config)#interface HundredGigabitEthernet1/0/1
Leaf_1(config-if)# description Spine_1
Leaf_1(config-if)# ip address 172.16.1.1 255.255.255.252
Leaf_1(config-if)# ip router isis
Leaf_1(config-if)# isis network point-to-point
This action will reset all neighbor connections on the interface.
Leaf_1(config-if)#exit
Leaf_1(config)#interface HundredGigabitEthernet1/0/2
Leaf_1(config-if)# description Spine_2
Leaf_1(config-if)# ip address 172.16.1.5 255.255.255.252
Leaf_1(config-if)# ip router isis
Leaf_1(config-if)# isis network point-to-point
This action will reset all neighbor connections on the interface.
Leaf_1(config-if)#exit
Leaf_1(config)#interface loopback1
Leaf_1(config-if)# ip address 10.0.0.1 255.255.255.255
Leaf_1(config-if)#exit
```

**⚠** В протоколе IS-IS рекомендуем задействовать режим p2p (point-to-point). Он используется при установлении соединений на линковочных сетях, где каждый маршрутизатор имеет только одного соседа. Режим point-to-point уменьшает количество служебных пакетов протокола IS-IS, ускоряет установление соседств, уменьшая время сходимости протокола.

Задействование протокола маршрутизации IS-IS:

```
Leaf_1(config)#router isis
Leaf_1(router-isis)# address-family ipv4 unicast
Leaf_1(router-isis-af)# redistribute connected
Leaf_1(router-isis-af)# exit
Leaf_1(router-isis)# net 49.0001.0001.0001.0001.00
Leaf_1(router-isis)#exit
```

Задействование протокола BGP.

**⚠** Поддержка протокола BGP предоставляется по лицензии (см. [Установка лицензий](#)).

```
Leaf_1(config)#router bgp 65500
Leaf_1(router-bgp)# bgp router-id 10.0.0.1
This action will reset all neighbor connections and clear BGP routing table.
Leaf_1(router-bgp)# address-family ipv4 unicast
Leaf_1(router-bgp-af)# exit
Leaf_1(router-bgp)# address-family l2vpn evpn
This action will reset all neighbor connections and clear BGP routing table.
Leaf_1(router-bgp-af)# exit
Leaf_1(router-bgp)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbrgrp)# remote-as 65500
This action will reset connection with all neighbors in peer group.
Leaf_1(router-bgp-nbrgrp)# update-source loopback 1
This action will reset connection with all neighbors in peer group.
Leaf_1(router-bgp-nbrgrp)# fall-over bfd
Leaf_1(router-bgp-nbrgrp)# exit
Leaf_1(router-bgp)# neighbor 10.0.1.1
Leaf_1(router-bgp-nbr)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbr)# address-family ipv4 unicast
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# exit
Leaf_1(router-bgp)# neighbor 10.0.2.2
Leaf_1(router-bgp-nbr)# peer-group SPINE_GROUP
Leaf_1(router-bgp-nbr)# address-family ipv4 unicast
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# address-family l2vpn evpn
This action will reset connection with the neighbor.
Leaf_1(router-bgp-nbr-af)# exit
Leaf_1(router-bgp-nbr)# exit
Leaf_1(router-bgp)#exit
```

Настройка остальных устройств Leaf в схеме выполняется аналогично, с внесением необходимых изменений согласно схемы и плану IP-адресации.

Изменению от устройства к устройству подлежат следующие параметры:

- Hostname устройства;
- Description на интерфейсах;
- IP-адреса интерфейсов (физических + loopback);
- IS-IS net идентификатор;
- BGP router-id;
- IP-адреса BGP-соседей.

### 7.3 Проверка настроек underlay

После выполнения вышеописанных настроек необходимо выполнить проверку установления соседства протоколов IS-IS, BGP и BFD.

Используемые команды:

```
show isis neighbors
show ip bgp neighbors
show ip bfd neighbors
```

Пример выполнения вышеуказанных show-команд на устройстве Spine\_1:

```
Spine_1#show isis neighbors
```

System Id	Interface	Type	SNPA	State	Holdtime (s)	Circuit Id
Leaf_1	hu1/0/1	L1-2	e0d9.e326.d600	Up	29	
Leaf_2	hu1/0/2	L1-2	e0d9.e3f8.6e00	Up	29	
Leaf_3	hu1/0/3	L1-2	e0d9.e3d7.ea80	Up	27	

```
Spine_1#show ip bgp neighbors
```

BGP neighbor	Remote AS	Router ID	State	Uptime	Hold Time	Keepalive
10.0.0.1	65500	10.0.0.1	ESTABLISHED	00,00:01:15	90	30
10.0.0.2	65500	10.0.0.2	ESTABLISHED	00,00:00:58	90	30
10.0.0.3	65500	10.0.0.3	ESTABLISHED	00,00:00:44	90	30

```
Spine_1#show ip bfd neighbors
```

Neighbor	Local	State	Last Down Diag
10.0.0.1	10.0.1.1	Up	No Diagnostic
10.0.0.2	10.0.1.1	Up	No Diagnostic
10.0.0.3	10.0.1.1	Up	No Diagnostic

Пример выполнения вышеуказанных show-команд на устройстве Leaf\_1:

```
Leaf_1#show isis neighbors
```

System Id	Interface	Type	SNPA	State	Holdtime (s)	Circuit Id
Spine_1	hu1/0/1	L1-2	cc9d.a253.d680	Up	29	
Spine_2	hu1/0/2	L1-2	e0d9.e317.6b40	Up	28	

```
Leaf_11#show ip bgp neighbors
```

BGP neighbor	Remote AS	Router ID	State	Uptime	Hold Time	Keepalive
10.0.1.1	65500	10.0.1.1	ESTABLISHED	00,00:01:25	90	30
10.0.2.2	65500	10.0.2.2	ESTABLISHED	00,00:01:23	90	30

```
Leaf_11#show ip bfd neighbors
```

Neighbor	Local	State	Last Down Diag
10.0.1.1	10.0.0.1	Up	No Diagnostic
10.0.2.2	10.0.0.1	Up	No Diagnostic

В случае успешного установления всех соседств в схеме должна быть обеспечена IP-связность между всеми устройствами. Проверить это можно, выполнив команду ICMP ping с любого устройства до любого другого, указав в качестве src и dst IP-адреса loopback-интерфейсов устройств.

Пример выполнения ping с Leaf\_1 до Leaf\_3:

```
Leaf_1#ping 10.0.0.3 source 10.0.0.1
Pinging 10.0.0.3 with 18 bytes of data:
```

```
18 bytes from 10.0.0.3: icmp_seq=1. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=2. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=3. time=0 ms
18 bytes from 10.0.0.3: icmp_seq=4. time=0 ms
```

```
----10.0.0.3 PING Statistics----
```

```
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms) min/avg/max = 0/0/0
```

## 7.4 Ожидаемый результат

- Все соседства протокола IS-IS и BFD в состоянии UP. Соседства протокола BGP в состоянии ESTABLISHED.
- IP-связность между всеми устройствами схемы установлена.

## 8 Балансировка трафика

### 8.1 Общие положения

На устройствах, поддерживающих VXLAN-инкапсуляцию, начиная с версии ПО 6.6.10, реализован механизм изменения UDP-порта источника во внешнем UDP-заголовке.

Для устройств MES5410-48 и MES5500-32 реализован механизм изменения порта per-packet (попакетно), для остальных устройств per-VTEP, т.е. один UDP-порт источника для всех пакетов до конкретного VTEP.

Этот механизм не требует дополнительной настройки.

До версии 6.6.10 во всех внешних UDP-заголовках UDP-порт источника имел значение 65535.

На устройствах уровня SuperSpine и Spine (на уровне Spine влияние будет только на восходящий трафик) рекомендуем использовать алгоритм балансировки src-dst-mac-ip-port.

В сумме эти меры расширяют возможности по балансировке инкапсулированного трафика.

Важно иметь в виду, что исходящий VTEP принимает решение о балансировке инкапсулированного пакета в сторону уровня Spine на основании данных из заголовков исходного пакета.

Инкапсуляция происходит уже после принятия решения о балансировке.

В случае балансировки трафика до удаленного multihomed-клиента, решение о балансировке также принимает исходящий VTEP. При этом он отвечает за распределение потоков трафика как между Spine, так и между Leaf назначения — участниками Ethernet-сегмента.

### 8.2 Параметры hash-функции

Начиная с версии ПО 6.6.11 добавлена возможность изменения некоторых параметров hash-функции, которая используется в механизме балансировки трафика. Настройка разных значений этих параметров на разных устройствах позволяет получать для одного и того же пакета разный результат на выходе hash-функции. Соответственно может быть принято решение о балансировке пакета в другой линк из состава LAG. Описанный механизм позволяет избежать эффекта поляризации трафика в случае следующих друг за другом LAG по пути движения пакетов.

Пример настройки данных параметров:

```
console(config)#port-channel load-balance hash salt 12345
console(config)#port-channel load-balance hash seed 54321
```

Оба параметра имеют числовое значение в диапазоне от 1 до 4294967295.

В случае балансировки трафика при ECMP поведение аналогичное. Изменяя параметры hash-функции мы можем влиять на балансировку трафика в пределах ECMP-группы. Для топологии Spine-Leaf рекомендуется использовать данные настройки на уровне SuperSpine.

## 9 Настройка overlay. VXLAN

Схема аналогична используемой в разделах [Настройка underlay с использованием протокола OSPF](#) и [Настройка underlay с использованием протокола IS-IS](#).

Перед настройкой VXLAN выполните [предварительную настройку](#) устройств и настройку согласно одному из вышеупомянутых разделов данного руководства.

### 9.1 Настройка VXLAN

 Поддержка VXLAN предоставляется по лицензии (см. [Установка лицензий](#)).

Создайте на устройстве Leaf\_1 VXLAN-инстанс с именем test\_vxlan. Установите для него значение VNI 101000 и привяжите VLAN 1000. Предварительно VLAN 1000 должна быть создана и присутствовать во VLAN database.

```
Leaf_1(config)#vlan database
Leaf_1(config-vlan)#vlan 1000
Leaf_1(config-vlan)#vxlan test_vxlan
Leaf_1(config-vxlan)#vni 101000
Leaf_1(config-vxlan)#vlan 1000
```

По умолчанию созданный VXLAN-инстанс находится в состоянии по shutdown, т.е. включен. В контексте настройки vxlan его можно выключить командой shutdown. Пример:


```
Leaf_1(config-vxlan)#shutdown
```


Создайте аналогичный VXLAN на устройствах Leaf\_2 и Leaf\_3.

VLAN 1000 в данном примере является клиентской VLAN. Необходимо сделать интерфейсы, ведущие в сторону Host1 и Host2 членами данной VLAN:

```
Leaf_1(config)#interface TenGigabitEthernet1/0/11
Leaf_1(config-if)#description Host1
Leaf_1(config-if)#switchport access vlan 1000

Leaf_3(config)#interface TenGigabitEthernet1/0/11
Leaf_3(config-if)#description Host2
Leaf_3(config-if)#switchport access vlan 1000
```

 В данном примере VXLAN-инстанс создается на всех устройствах Leaf в учебных целях для повышения наглядности и информативности выводов show-команд. В реальной IP-фабрике VXLAN создаются по необходимости и на определенных устройствах Leaf.

 При использовании номеров автономных систем < 65535 есть возможность привязки разных VLAN к одной и той же VXLAN на разных VTEP без дополнительных настроек. В случае номеров автономных систем > 65535 для использования разных VLAN необходимо задействовать настройку RT – **route-target**. Её описание приведено далее по тексту.

Использование расширенного community route-target позволяет использовать разные VLAN на разных VTEP для AS >65535.

Пример конфигурации VXLAN для трёх Leaf с использованием RT и различных номеров VLAN (№ AS > 65535):

```
Leaf_1:

vxlan test_vxlan
 vni 101000
 vlan 1000
 route-target both 65600:100
 route-target both 65600:200
 exit

Leaf_2:

vxlan test_vxlan
 vni 101000
 vlan 500
 route-target both 65600:100
 exit


Leaf_3:

vxlan test_vxlan
 vni 101000
 vlan 200
 route-target both 65600:200
 exit
```

На всех трёх Leaf используются разные номера VLAN. Благодаря указанию RT Leaf\_1 установит двунаправленные VXLAN туннели до Leaf\_2 и до Leaf\_3.

При этом между Leaf\_2 и Leaf\_3 туннель установлен не будет в силу разности VLAN, привязанных к VXLAN, и разности RT.

Для связности всех Leaf в VXLAN test\_vxlan «каждый с каждым» необходимо использовать одинаковые RT на всех Leaf.

 Максимальное количество назначенных RT в одном экземпляре VXLAN — два.

## 9.2 Проверка настройки VXLAN

Проконтролировать создание VXLAN можно несколькими способами:

- используя специальные show-команды;
- в информации протокола BGP;
- в выводе текущей конфигурации.

### 9.2.1 Show-команды

Посмотреть информацию, в том числе и детальную о созданных VXLAN можно при помощи специальных команд:

- show vxlan — отображает информацию обо всех созданных экземплярах VXLAN в табличном виде;
- show vxlan WORD<1-64> — отображает детальную информацию об определенной VXLAN;
- show vxlan tunnels — отображает все установленные VXLAN-туннели;
- show vxlan tunnels WORD<1-64> — отображает установленные VXLAN-туннели для определенной VXLAN.

Примеры вывода информации вышеописанных команд представлены ниже:

```
Leaf_1#show vxlan
```

Name	VNI	VLAN ID	Status admin/oper	BUM Forwarding	Route Distinguisher
test_vxlan	101000	1000	UP/UP	Ingress Repl.	10.0.0.1:1000

```
Leaf_1#show vxlan test_vxlan
```

```
test_vxlan
  VxLAN Network ID is 101000, VLAN ID is 1000
  Administrative status is UP
  Operational status is UP

  Local Router ID is 10.0.0.1
  Route Distinguisher is 10.0.0.1:1000 (auto-assigned)
  Route Target is: 65535:268536456 (auto-assigned)

  Broadcast/Unknown Unicast/Multicast traffic
  is forwarded in Ingress Replication mode
```

```
Leaf_1#show vxlan tunnels
```

```
test_vxlan
```

Destination	Source	Status
101000:10.0.0.2	101000:10.0.0.1	Up
101000:10.0.0.3	101000:10.0.0.1	Up

```
Leaf_1#show vxlan tunnels test_vxlan
```

```
test_vxlan
```

Destination	Source	Status
101000:10.0.0.2	101000:10.0.0.1	Up
101000:10.0.0.3	101000:10.0.0.1	Up

## 9.2.2 Вывод информации протокола BGP

```
Leaf_1#show ip bgp l2vpn evpn
```

```
BGP table version is 9, local router ID is 10.0.0.1
Status codes: * - valid, > - best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Nexthop	Metric	LocPrf	Weight	Path
Route distinguisher: 10.0.0.1:1000					
*> [3][0][32][10.0.0.1]/88	0.0.0.0	0	100	0	?

```
Route distinguisher: 10.0.0.2:1000
```

```

*>i[3][0][32][10.0.0.2]/88
      10.0.0.2      0      100      0      ?
Route distinguisher: 10.0.0.2:1000
* i[3][0][32][10.0.0.2]/88
      10.0.0.2      0      100      0      ?
Route distinguisher: 10.0.0.3:1000
*>i[3][0][32][10.0.0.3]/88
      10.0.0.3      0      100      0      ?
Route distinguisher: 10.0.0.3:1000
* i[3][0][32][10.0.0.3]/88
      10.0.0.3      0      100      0      ?

```

Рассмотрим первую запись:

Network	Nexthop	Metric	LocPrf	Weight	Path
Route distinguisher: 10.0.0.1:1000					
*> [3][0][32][10.0.0.1]/88	0.0.0.0	0	100	0	?

Где:

- [ 3 ] – тип маршрута;
- [ 0 ] – EthTag. В текущей версии ПО не используется;
- [ 32 ] – длина IP-адреса источника маршрута;
- [ 10.0.0.1 ] – IP-адрес источника маршрута;
- [ 88 ] – полная длина маршрута.

Наличие этой записи говорит о создании локального маршрута типа 3, необходимого для приема BUM-трафика методом ingress-replication. Также данный маршрут анонсируется всем BGP-соседям с включенной AF I2vpn evpn.

Остальные записи говорят о наличии маршрутов типа 3 до удаленных VTEP.

Количество маршрутов по 2 экземпляра, т.к. имеется 2 альтернативных пути до каждого другого Leaf через два Spine.

Кроме того, информацию о туннелях, предназначенных для распространения BUM-трафика, можно увидеть в выводе команды **show evpn inclusive-multicast**. Пример вывода:

```

Leaf_1#show evpn inclusive-multicast

VXLAN test_vxlan

IP Address      VNI      Source Tunnel Address  Remote Route Distinguisher  Tunnel Type
-----
10.0.0.1        101000   Local  10.0.0.1           10.0.0.1:1000   Ingress Repl.
10.0.0.2        101000   Remote 10.0.0.2           10.0.0.2:1000   Ingress Repl.
10.0.0.3        101000   Remote 10.0.0.3           10.0.0.3:1000   Ingress Repl.

```

### 9.2.3 Проверка работоспособности VXLAN

В качестве проверки работоспособности созданной VXLAN можно использовать проверку IP-связности между Host1 и Host2. Для этого их IP-интерфейсы должны быть в одной подсети.

После успешного выполнения команды ping проконтролируйте наличие изученных MAC-адресов в таблицах Leaf\_1 и Leaf\_3. Пример:

```
Leaf_1#show mac address-table
Flags: I - Internal usage VLAN
Aging time is 300 sec
```

Vlan	Mac Address	Interface	Type
1	e0:d9:e3:26:d6:00	0	self
1000	0c:9d:92:61:9f:c4	te1/0/11	dynamic
1000	e0:d9:e3:a8:45:40	10.0.0.3	evpn-vxlan
hu1/0/1(I)	cc:9d:a2:53:d6:80	hu1/0/1	dynamic
hu1/0/1(I)	cc:9d:a2:53:d6:81	hu1/0/1	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:40	hu1/0/2	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:41	hu1/0/2	dynamic

```
Leaf_3#show mac address-table
Flags: I - Internal usage VLAN
Aging time is 300 sec
```

Vlan	Mac Address	Interface	Type
1	e0:d9:e3:d7:ea:80	0	self
1000	0c:9d:92:61:9f:c4	10.0.0.1	evpn-vxlan
1000	e0:d9:e3:a8:45:40	te1/0/11	dynamic
hu1/0/1(I)	cc:9d:a2:53:d6:80	hu1/0/1	dynamic
hu1/0/1(I)	cc:9d:a2:53:d6:83	hu1/0/1	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:40	hu1/0/2	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:43	hu1/0/2	dynamic

MAC-адрес Host1 должен быть в таблице Leaf\_3, тип evpn-vxlan. Аналогично MAC-адрес Host2 должен быть в таблице Leaf\_1, тип evpn-vxlan.

В выводе информации протокола BGP можно наблюдать маршрут типа 2 с указанием изученного MAC-адреса удаленного хоста в качестве адреса назначения маршрута:

```
Leaf_1#show ip bgp l2vpn evpn

BGP table version is 14, local router ID is 10.0.0.1
Status codes: * - valid, > - best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Nexthop	Metric	LocPrf	Weight	Path
Route distinguisher: 10.0.0.1:1000					
*> [2][0][0][48][0c:9d:92:61:9f:c4][0][0.0.0.0]/216	0.0.0.0	0	100	0	?
Route distinguisher: 10.0.0.3:1000					
*>i[2][0][0][48][e0:d9:e3:a8:45:40][0][0.0.0.0]/216	10.0.0.3	0	100	0	?

```

Route distinguisher: 10.0.0.3:1000
* i [2] [0] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216
      10.0.0.3      0      100      0      ?

Route distinguisher: 10.0.0.1:1000
*> [3] [0] [32] [10.0.0.1] /88
      0.0.0.0      0      100      0      ?

Route distinguisher: 10.0.0.2:1000
*>i [3] [0] [32] [10.0.0.2] /88
      10.0.0.2      0      100      0      ?

Route distinguisher: 10.0.0.2:1000
* i [3] [0] [32] [10.0.0.2] /88
      10.0.0.2      0      100      0      ?

Route distinguisher: 10.0.0.3:1000
*>i [3] [0] [32] [10.0.0.3] /88
      10.0.0.3      0      100      0      ?

Route distinguisher: 10.0.0.3:1000
* i [3] [0] [32] [10.0.0.3] /88
      10.0.0.3      0      100      0      ?

```

Пример записи:

Network	Nexthop	Metric	LocPrf	Weight	Path
Route distinguisher: 10.0.0.3:1000					
*>i [2] [0] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216					
	10.0.0.3	0	100	0	?

Где:

- [2] – тип маршрута;
- [0] – ESI (Ethernet segment identifier);
- [0] – EthTag. В текущей версии ПО не используется;
- [48] – длина MAC-адреса;
- [e0:d9:e3:a8:45:40] – MAC-адрес, изученный на удаленном VTEP;
- [0] – длина IP-адреса;
- [0.0.0.0] – IP-адрес;
- 216 – полная длина маршрута.

Эта запись говорит о наличии маршрута типа 2. На удаленном VTEP изучен MAC-адрес e0:d9:e3:a8:45:40. Для отправки пакетов по этому маршруту необходимо использовать nexthop 10.0.0.3.

Также основную информацию об изученных в VXLAN MAC-адресах можно получить в выводе show-команды **show evpn mac-ip**:

```
Leaf_1#show evpn mac-ip

VXLAN test_vxlan

VNI          VLAN MAC Address      IP          ESI          Next Hop
-----
101000      1000 0c:9d:92:61:9f:c4 -            -            te1/0/11
101000      1000 e0:d9:e3:a8:45:40 -            -            10.0.0.3
```

### **Зеркалирование трафика**

Зеркалирование трафика можно использовать как инструмент контроля на любом устройстве схемы и на любом этапе. Но в случае с инкапсулированным в VXLAN трафиком необходимо руководствоваться следующей информацией: исходящий трафик перехватывается зеркалом до инкапсуляции, поэтому трафик, упакованный в VXLAN, необходимо зеркалировать только на устройстве, для которого он является входящим.

## **9.3 ARP suppression**

Функция ARP suppression позволяет уменьшать BUM-трафик в сети путем уменьшения количества пересылаемых через IP-фабрику пакетов ARP.

Включается командой в контексте vxlan:

```
Leaf_1#configure
Leaf_1(config)#vxlan test_vxlan
Leaf_1(config-vxlan)#arp-suppression
```

Работу данной функции обеспечивает специальный кэш, в котором хранятся соответствия MAC- и IP-адресов. Пример вывода содержимого данного кэша:

```
Leaf_1#show arp suppression-cache

Total number of entries: 2
ARP suppression-cache timeout is 300 sec

IP address      Vtep      MAC address      VLAN      Port      Flags      Age
-----
192.168.13.1    0.0.0.0    e0:d9:e3:d7:ea:80 1000      te1/0/11  local      00:00:33
192.168.13.2    10.0.0.3   e0:d9:e3:26:d6:00 1000      0         remote     00:00:08
```

Записи в кэш добавляются:

- При получении ARP request в привязанной к VXLAN VLAN (записи типа local);
- При получении BGP update с маршрутом типа 2 MAC-IP (записи типа remote).

Принцип работы: Host1 посылает ARP-запрос, чтобы узнать MAC-адрес Host2. Если Leaf\_1 уже знает связку MAC-IP Host2 (в кэше есть соответствующая запись), то Leaf\_1 сам ответит на ARP запрос от имени Host2, не пересылая его через IP-фабрику.

Записи удаляются из кэша:

- При получении BGP update с withdraw типа 2 MAC-IP (для записей типа remote);
- В момент удаления MAC-адреса с устройства соответствующая запись типа local переходит в тип *local, inactive*. В этот момент отправляется BGP update с withdraw соответствующей связки MAC-IP. Запись снова может стать активной при повторном изучении MAC-адреса. Если этого не произошло, запись будет удалена по истечении половины suppression-cache timeout;
- По истечении suppression-cache timeout.

Таймер suppression-cache timeout устанавливает максимальное время жизни записей типа local в таблице arp suppression-cache:

```
Leaf_2(config)#arp suppression-cache timeout  
<30-40000000> Seconds
```

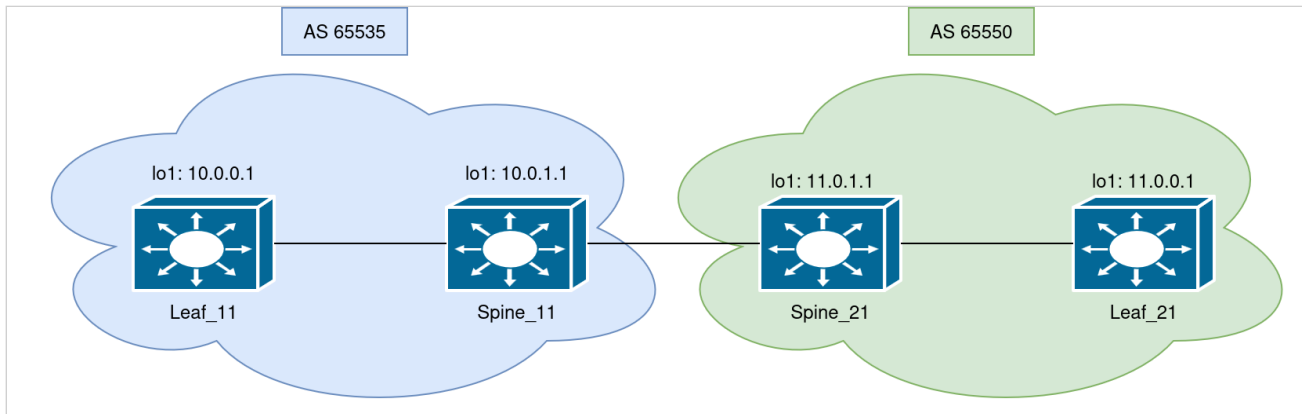
Значение по умолчанию 5 минут.

#### 9.4 Ожидаемый результат

- VXLAN успешно создается;
- Необходимую информацию о VXLAN и об изученных MAC-адресах можно посмотреть в выводе соответствующих show-команд;
- Есть возможность передачи пользовательского трафика между портами разных Leaf через VXLAN-туннель;
- Функция ARP suppression позволяет уменьшить количество пересылаемых пакетов протокола ARP.

## 10 Работа EVPN/VXLAN поверх eBGP

При использовании eBGP есть необходимость в дополнительных настройках. Рассмотрим в качестве примера упрощенную схему, подразумевающую прямое соединение двух POD между собой.



В случае передачи маршрута eBGP-соседу передающий маршрутизатор должен указать в поле Next hop в BGP-update свой адрес. Это делает невозможным установление VXLAN-туннелей, которые строятся между loopback интерфейсами Leaf.

Чтобы не менять значение поля Next hop в этом случае необходимо использовать настройку **next-hop-unchanged**.

Пример настройки Spine\_11 в части BGP:

```
router bgp 65535
...
neighbor 11.0.1.1
  remote-as 65550
  update-source loopback 1
  address-family ipv4 unicast
  exit
!
address-family l2vpn evpn
  next-hop-unchanged
  exit
exit
```

Данная настройка используется в настройках конкретного соседа в контексте AF l2vpn evpn. Во всех маршрутах, передаваемых данному соседу в рамках AF l2vpn evpn, поле Next hop в BGP-update заменяться не будет.

Это позволит доставить BGP-update до всех Leaf другой AS в неизменном виде.

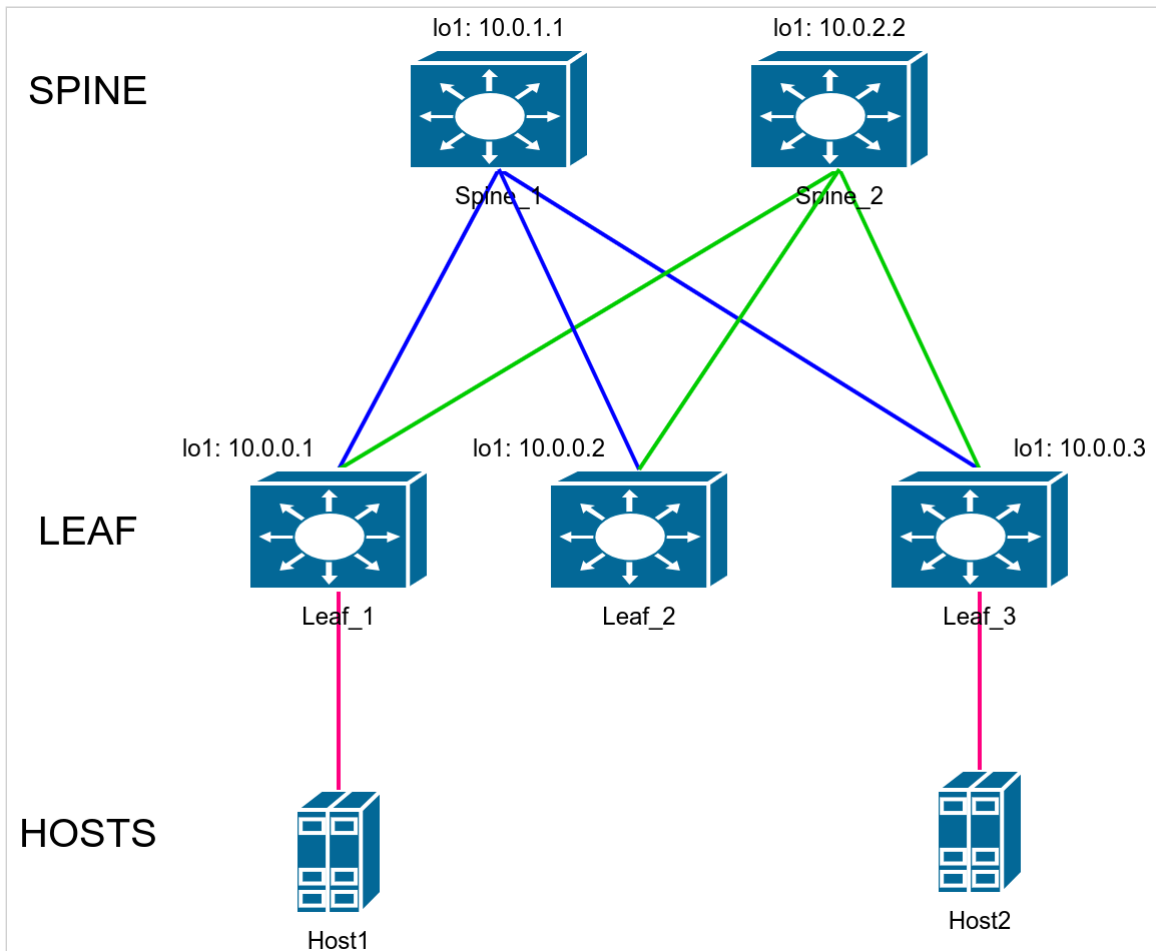
Поскольку теперь VTEP находятся в разных AS, автоматическая генерация RT, опирающаяся на номер AS, даст разные значения на разных VTEP. Чтобы это устранить, необходимо **ручное назначение RT** в VXLAN. Пример настройки VXLAN для этого случая на обоих Leaf:

```
vxlan test_vxlan
vni 101000
vlan 1000
route-target both 1:1000
exit
!
```

В комплексе эти 2 настройки позволят корректно распространяться EVPN маршрутам между AS и устанавливать VXLAN-туннели.

## 11 Настройка overlay. Multicast VXLAN

Используемая схема аналогична разделу [Настройка overlay. VXLAN](#).



В качестве отправных конфигураций устройств в данном разделе используются конфигурации из приложения [Конфигурации с использованием протокола OSPF](#).

Дополнительные настройки, необходимые для работы multicast vxlan, описаны далее в данном разделе.

В [приложении 1](#) есть полные конфигурации устройств, получаемые по окончании данного раздела.

### 11.1 Настройка Spine

Настройка протокола PIM:

```
Spine_1(config)#interface range HundredGigabitEthernet1/0/1-3
Spine_1(config-if-range)#ip pim
Spine_1(config-if-range)#exit
Spine_1(config)#interface loopback2
Spine_1(config-if)# ip address 10.100.100.100 255.255.255.255
Spine_1(config-if)# description RP_IP
Spine_1(config-if)#exit
Spine_1(config)#ip multicast-routing pim
Spine_1(config)#ip pim rp-address 10.100.100.100
```

**⚠** RP-адрес на всех устройствах Spine одинаковый.

## Настройка протокола MSDP:

```
Spine_1(config)#router msdp
Spine_1(config-msdp)#connect-source 10.0.1.1
Spine_1(config-msdp)#originator-ip 10.100.100.100
Spine_1(config-msdp)#peer 10.0.2.2
Spine_1(config-peer)#mesh-group TESTGR
Spine_1(config-peer)#exit
Spine_1(config-msdp)#exit
```

Где:

- connect-source – исходящий адрес для установления соединения с MSDP-пирами (адрес loopback 1 на Spine);
- originator-ip – IP-адрес, используемый в качестве адреса RP в source-active сообщениях (адрес loopback 2 на Spine);
- mesh-group – имя полносвязной группы. Должно быть одинаковым для всех Spine фабрики.

Настройка остальных устройств Spine в схеме выполняется аналогично, с внесением необходимых изменений согласно схемы и плану IP-адресации.

Изменению от устройства к устройству подлежат следующие параметры:

- connect-source протокола MSDP;
- IP-адрес peer-протокола MSDP.


## 11.2 Настройка Leaf

Настройка протокола PIM:

```
Leaf_1(config)#interface range HundredGigabitEthernet1/0/1-2
Leaf_1(config-if-range)#ip pim
Leaf_1(config-if-range)#exit
Leaf_1(config)#interface loopback 1
Leaf_1(config-if)#ip pim
Leaf_1(config-if)#exit
Leaf_1(config)#ip multicast-routing pim
Leaf_1(config)#ip pim rp-address 10.100.100.100
Leaf_1(config)#ip multicast multipath group-paths-num
```

Настройка остальных устройств Leaf в схеме выполняется аналогично. Отличий в блоке команд, приведенном выше, от устройства к устройству нет.

Стоит отметить назначение команды **ip multicast multipath group-paths-num**. Эта команда включает балансировку пакетов PIM Join в сторону доступных RP. Кроме того она задействует метод балансировки, при котором хеш-функция, подсчитанная на основе адреса группы, делится по модулю на N, где N – количество доступных RP.

 Вышеуказанный метод необходим для корректной работы балансировки при использовании EVPN/VXLAN. На практике он приводит к «синхронизации» VTEP и выбору одного и того же RP для отправки трафика конкретной группы.

Более подробно рассмотрим результат работы балансировки в следующем разделе.

### 11.3 Настройка multicast vxlan

Режим работы VXLAN multicast предоставляет возможность автоматического поиска удаленных VTEP. Репликация BUM-трафика осуществляется не на исходящем VTEP, как при дефолтном режиме работы VXLAN ingress replication, а посредством PIM multicast, ближе к точке назначения. Применимо к обсуждаемой топологии это означает репликацию не на исходящих Leaf, а на Spine-коммутаторах.

Создадим на устройстве Leaf\_1 4 разных VXLAN-инстанса.

Пример настройки:

```
Leaf_1(config)#vlan database
Leaf_1(config-vlan)#vlan 2-5
Leaf_1(config-vlan)#exit
Leaf_1(config)#vxlan mcast2
Leaf_1(config-vxlan)# vni 102
Leaf_1(config-vxlan)# vlan 2
Leaf_1(config-vxlan)# mcast-group 233.0.0.2
Leaf_1(config-vxlan)#exit
Leaf_1(config)#vxlan mcast3
Leaf_1(config-vxlan)# vni 103
Leaf_1(config-vxlan)# vlan 3
Leaf_1(config-vxlan)# mcast-group 233.0.0.3
Leaf_1(config-vxlan)#exit
Leaf_1(config)#vxlan mcast4
Leaf_1(config-vxlan)# vni 104
Leaf_1(config-vxlan)# vlan 4
Leaf_1(config-vxlan)# mcast-group 233.0.0.4
Leaf_1(config-vxlan)#exit
Leaf_1(config)#vxlan mcast5
Leaf_1(config-vxlan)# vni 105
Leaf_1(config-vxlan)# vlan 5
Leaf_1(config-vxlan)# mcast-group 233.0.0.5
Leaf_1(config-vxlan)#exit
```

Команда **mcast-group** <IP> в контексте VXLAN включает репликацию BUM-трафика в данной VXLAN посредством PIM multicast.

BUM-трафик, пришедший на VLAN VXLAN инстанса, перехватывается на CPU и инкапсулируется в сообщение PIM Register для регистрации на RP.

После регистрации создаётся VXLAN-туннель в сторону RP и BUM-трафик отправляется инкапсулированным в VXLAN-заголовок с multicast destination ip в UDP-заголовке.

- ⚠ Все Leaf в пределах одной и той же VXLAN должны использовать один и тот же метод репликации (и один и тот же адрес группы в случае multicast-репликации).  
Одна mcast-group может быть назначена на несколько VXLAN instance. На данный момент доступно 256 уникальных multicast-групп.  
Команда **mcast-group** <IP> отключает отправку EVPN маршрутов типа 3 в пределах соответствующей VXLAN.

Настройка интерфейса loopback:

```
Leaf_1(config)#interface loopback1
Leaf_1(config-if)#ip igmp static-group 233.0.0.2
Leaf_1(config-if)#ip igmp static-group 233.0.0.3
Leaf_1(config-if)#ip igmp static-group 233.0.0.4
Leaf_1(config-if)#ip igmp static-group 233.0.0.5
```

Команда **ip igmp static-group** позволяет Leaf подписаться на указанную группу и начать её слушать. В этот момент в сторону RP отправляется сообщение join на эту группу с учетом механизма балансировки, включаемого командой **ip multicast multipath group-paths-num**. Сообщения join распределяются между RP. На каждом Spine (RP) создаются записи (\*, G) для соответствующих групп. На практике в используемой схеме это будет выглядеть так:

```
Spine_1#show ip mroute
IP Multicast Routing Table

Flags: D - Dense, S - Sparse, X - IGMP Proxy, s - SSM Group,
       C - Connected, L - Local, R - RP-bit set, F - Register flag,
       T - SPT-bit set, I - Received Source Specific Host Report
Timers: Uptime/Expires

(*, 233.0.0.3), uptime: 00:00:04, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1

(*, 233.0.0.5), uptime: 00:00:03, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1
```

Устройство Leaf\_1 отправило в сторону Spine\_1 PIM join сообщения на группы 233.0.0.3 и 233.0.0.5. Сообщения на группы 233.0.0.2 и 233.0.0.4 были отправлены в сторону Spine\_2:

```
Spine_2#show ip mroute
IP Multicast Routing Table

Flags: D - Dense, S - Sparse, X - IGMP Proxy, s - SSM Group,
       C - Connected, L - Local, R - RP-bit set, F - Register flag,
       T - SPT-bit set, I - Received Source Specific Host Report
Timers: Uptime/Expires

(*, 233.0.0.2), uptime: 00:00:09, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1

(*, 233.0.0.4), uptime: 00:00:07, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1
```

Таким образом трафик различных групп будет поступать на Leaf через разные Spine, тем самым обеспечивая распределение нагрузки.

Настроим другие Leaf точно таким же образом.

Проверим теперь таблицы multicast-маршрутов на обоих Spine:

```
Spine_1#show ip mroute
IP Multicast Routing Table

Flags: D - Dense, S - Sparse, X - IGMP Proxy, s - SSM Group,
       C - Connected, L - Local, R - RP-bit set, F - Register flag,
       T - SPT-bit set, I - Received Source Specific Host Report
Timers: Uptime/Expires

(*, 233.0.0.3), uptime: 00:00:37, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1, hu1/0/2, hu1/0/3
```

```
(* , 233.0.0.5), uptime: 00:00:37, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1, hu1/0/2, hu1/0/3
```

```
Spine_2#sh ip mroute
IP Multicast Routing Table
```

```
Flags: D - Dense, S - Sparse, X - IGMP Proxy, s - SSM Group,
       C - Connected, L - Local, R - RP-bit set, F - Register flag,
       T - SPT-bit set, I - Received Source Specific Host Report
```

```
Timers: Uptime/Expires
```

```
(* , 233.0.0.2), uptime: 00:01:39, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1, hu1/0/2, hu1/0/3
```

```
(* , 233.0.0.4), uptime: 00:01:39, expires: never, RP 10.100.100.100, Flags: S
  Incoming interface: Null, RPF neighbor 10.100.100.100
  Outgoing interface list: hu1/0/1, hu1/0/2, hu1/0/3
```

После настройки других Leaf аналогичным образом, они также отправили сообщения PIM join, распределив их между Spine (между RP). Таким образом интерфейсы hu1/0/2 и hu1/0/3 на обоих Spine были добавлены в Outgoing interface list обеих групп.

Осталось настроить порты на коммутаторах Leaf для подключения клиентских устройств:

```
Leaf_1(config)#interface TenGigabitEthernet1/0/9
Leaf_1(config)#description Host1_mcast
Leaf_1(config-if)#switchport mode trunk
Leaf_1(config-if)#switchport trunk allowed vlan add 2-5
Leaf_1(config-if)#switchport forbidden default-vlan

Leaf_3(config)#interface TenGigabitEthernet1/0/9
Leaf_3(config-if)#switchport mode trunk
Leaf_3(config-if)#switchport trunk allowed vlan add 2-5
Leaf_3(config-if)#switchport forbidden default-vlan
```

## 11.4 Проверка настройки multicast vxlan

Для создания multicast vxlan туннелей необходимо отправить BUM-трафик в клиентские VLAN. Отправим broadcast во VLAN 2-5 в клиентский порт на Leaf\_1.

Результат создания этих туннелей в show-команде **show vxlan tunnels** можно будет видеть на Leaf\_2 и Leaf\_3, т.к. в данном опыте broadcast однонаправленный и только Leaf\_2 и Leaf\_3 будут знать о существовании другого VTEP в сети (Leaf\_1).

В выводе соответствующих show-команд можно будет увидеть не только результат создания VXLAN локально на VTEP, но и результат создания туннелей.

Команды с примерами их выводов приведены ниже.

- show vxlan — отображает информацию обо всех созданных экземплярах VXLAN в табличном виде;
- show vxlan WORD<1-64> — отображает детальную информацию об определенной VXLAN;
- show vxlan tunnels — отображает все установленные VXLAN-туннели;
- show vxlan tunnels WORD<1-64> — отображает установленные VXLAN-туннели для определенной VXLAN.

Примеры вывода:

```
Leaf_3#show vxlan
```

Name	VNI	VLAN ID	Status admin/oper	BUM Forwarding	Route Distinguisher
test_vxlan	101000	1000	UP/UP	Ingress Repl.	10.0.0.3:1000
mcast2	102	2	UP/UP	Multicast VxLAN	10.0.0.3:2
mcast3	103	3	UP/UP	Multicast VxLAN	10.0.0.3:3
mcast4	104	4	UP/UP	Multicast VxLAN	10.0.0.3:4
mcast5	105	5	UP/UP	Multicast VxLAN	10.0.0.3:5

```
Leaf_3#show vxlan mcast2
```

```
mcast2
VxLAN Network ID is 102, VLAN ID is 2
Administrative status is UP
Operational status is UP

Local Router ID is 10.0.0.3
Route Distinguisher is 10.0.0.3:2 (auto-assigned)
Route Target is: 65500:268435558 (auto-assigned)

Broadcast/Unknown Unicast/Multicast traffic
is forwarded in multicast vxlan mode
Multicast Group address is 233.0.0.2
```

```
Leaf_3#show vxlan tunnels
```

```
test_vxlan
```

Destination	Source	Status
101000:10.0.0.1	101000:10.0.0.3	Up
101000:10.0.0.2	101000:10.0.0.3	Up

```
mcast2
```

Destination	Source	Status
-------------	--------	--------

```

102:10.0.0.1          102:10.0.0.3          Up
mcast3
Destination          Source                  Status
-----
103:10.0.0.1          103:10.0.0.3          Up
mcast4
Destination          Source                  Status
-----
104:10.0.0.1          104:10.0.0.3          Up
mcast5
Destination          Source                  Status
-----
105:10.0.0.1          105:10.0.0.3          Up
Leaf_3#show vxlan tunnels mcast2
mcast2
Destination          Source                  Status
-----
102:10.0.0.1          102:10.0.0.3          Up

```

Отправляемый на вход Leaf\_1 broadcast-трафик можно наблюдать на выходе Leaf\_3 в неизменном виде.

Для повышения наглядности опыта, демонстрирующего способ репликации BUM-трафика, рекомендуется оставить вещание одной VXLAN multicast-группы, т.е. отправлять клиентский broadcast в одной VLAN.

Посмотрим утилизацию портов на Leaf\_1:

```

Leaf_1#show int utilization
Port      Period, s  Sent, Kbit/s  Recv, Kbit/s  Frames sent  Frames rcv
-----
te1/0/9   15         0              1103          0             15029
...
hu1/0/1   15         4              4             115           112
hu1/0/2   15         1476           4             15145        111

```

Отправляемый трафик больше, чем входящий, по причине добавления дополнительного заголовка при инкапсуляции пакетов.

Далее видно, что трафик этой группы следует в сторону Spine\_2. Посмотрим утилизацию его интерфейсов:

```

Spine_2#sh int utilization
Port      Period, s  Sent, Kbit/s  Recv, Kbit/s  Frames sent  Frames rcv
-----
hu1/0/1   15         4              1476          113           15312
hu1/0/2   15         1476           4             15310        115
hu1/0/3   15         1476           4             15310        114

```

Как видно, копирование трафика в сторону других Leaf происходит на Spine\_2.

### **Зеркалирование трафика**

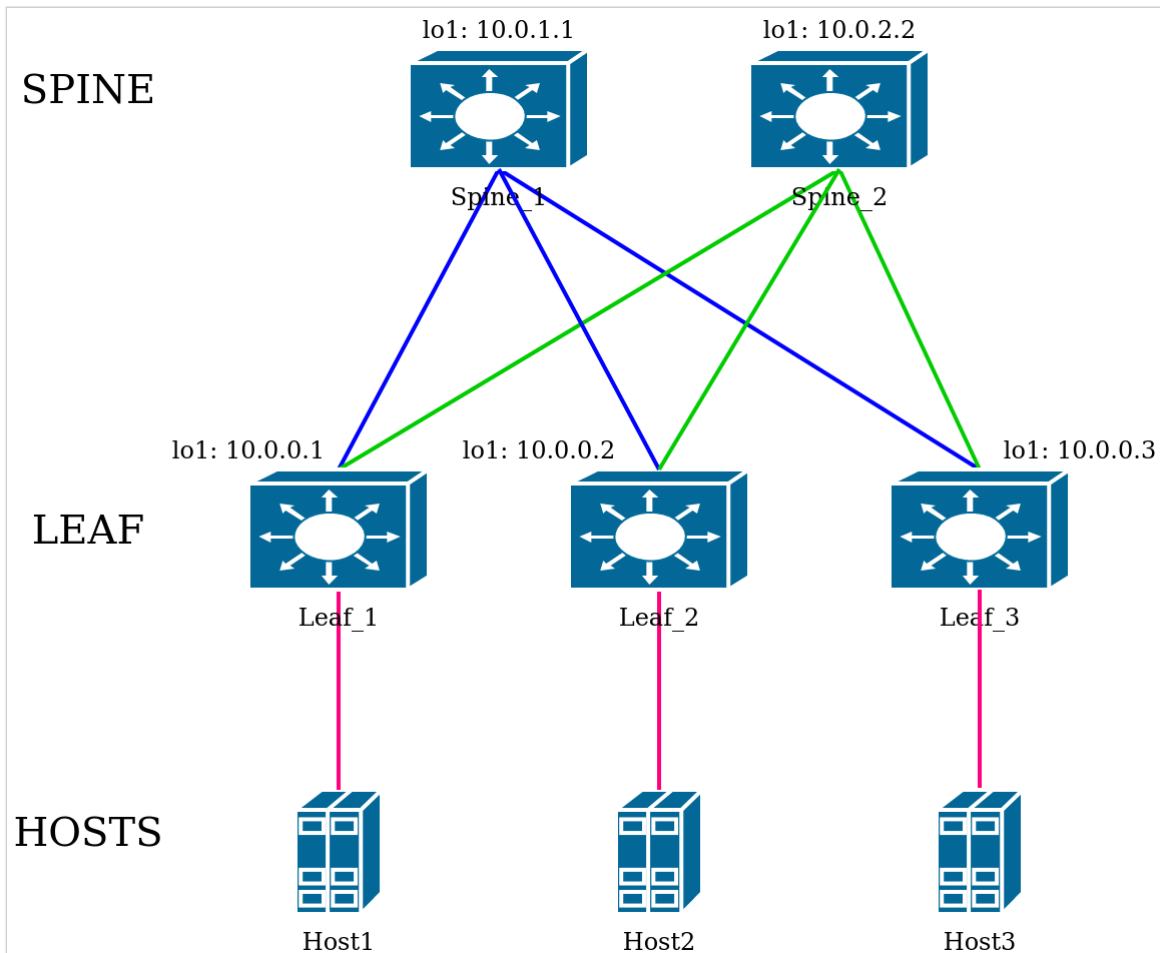
Зеркалирование трафика можно использовать как инструмент контроля на любом устройстве схемы и на любом этапе. Но в случае с инкапсулированным в VXLAN трафиком необходимо руководствоваться следующей информацией: исходящий трафик перехватывается зеркалом до инкапсуляции, поэтому трафик, упакованный в VXLAN, необходимо зеркалировать только на устройстве, для которого он является входящим.

## **11.5 Ожидаемый результат**

- Режим работы PIM multicast успешно задействуется;
- Информацию о режиме работы и об установленных таким образом туннелях можно посмотреть в выводе соответствующих show-команд;
- Репликация BUM-трафика посредством PIM multicast осуществляется на Spine-коммутаторах.

## 12 Настройка overlay. Symmetric IRB

Используемая схема представлена ниже.



В качестве отправных конфигураций устройств в данном разделе используются конфигурации из приложения [Конфигурации с использованием протокола OSPF без созданных VXLAN](#).

Дополнительные настройки, необходимые для работы L3VNI, описаны далее в данном разделе.

В [приложении 1](#) есть полные конфигурации устройств, получаемые по окончании данного раздела.

### 12.1 Настройка symmetric IRB

Для EVPN/VXLAN поддержан способ организации L3VPN с использованием Symmetric IRB, в частности маршрутизация через L3VNI.

Создадим на всех Leaf схемы L3VNI и подготовим его к работе.

Пример выполнения команд на Leaf\_1 для создания и подготовки L3VNI:

```
Leaf_1(config)#vlan database
Leaf_1(config-vlan)#vlan 100
Leaf_1(config-vlan)#exit
Leaf_1(config)#ip vrf VRF1
Leaf_1(config-vrf)#vni 100100
Leaf_1(config-vrf)#route-target both 65500:100100
Leaf_1(config-vrf)#exit
Leaf_1(config)#vxlan L3_vxlan
Leaf_1(config-vxlan)#vni 100100 ip-routing
Leaf_1(config-vxlan)#vlan 100
Leaf_1(config-vxlan)#exit
```

```
Leaf_1(config)#interface vlan 100
Leaf_1(config-if)#ip vrf VRF1
Leaf_1(config-if)#exit
```


Где:

- vni 100100 – L3VNI. Должен быть создан на всех VTEP, на которых необходима маршрутизация. К одной VRF может быть привязан только один L3VNI. Значения L3VNI не должны повторяться в различных VRF;
- route-target both 65500:100100 – задает значение RT community. Это расширенное community добавляется к отправляемым сообщениям update с маршрутами типа 2 (ip-mac) и 5. В свою очередь при приеме таких маршрутов по его значению определяется, в какую VRF устанавливаются принятые маршруты. Значения RT не должны повторяться в различных VRF;
- vxlan L3\_vxlan – VXLAN, необходимая для активации маршрутизации в L3VNI;
- vni 100100 ip-routing – настройка, указывающая, что vni 100100 будет использоваться для маршрутизации;
- vlan 100 – совместно с L3VNI необходимо использовать VLAN. Эта VLAN не будет использоваться для передачи какого-либо трафика, т.е. привязка её к какому-либо интерфейсу не обязательна. Обязательной является привязка интерфейса данной VLAN к соответствующей VRF.

Донастроим протокол BGP:

```
Leaf_1(config)#router bgp
Leaf_1(router-bgp)#vrf VRF1
Leaf_1(router-bgp-vrf)#address-family ipv4 unicast
Leaf_1(router-bgp-af-vrf)#redistribute connected
Leaf_1(router-bgp-af-vrf)#exit
Leaf_1(router-bgp-vrf)#exit
Leaf_1(router-bgp)#exit
```

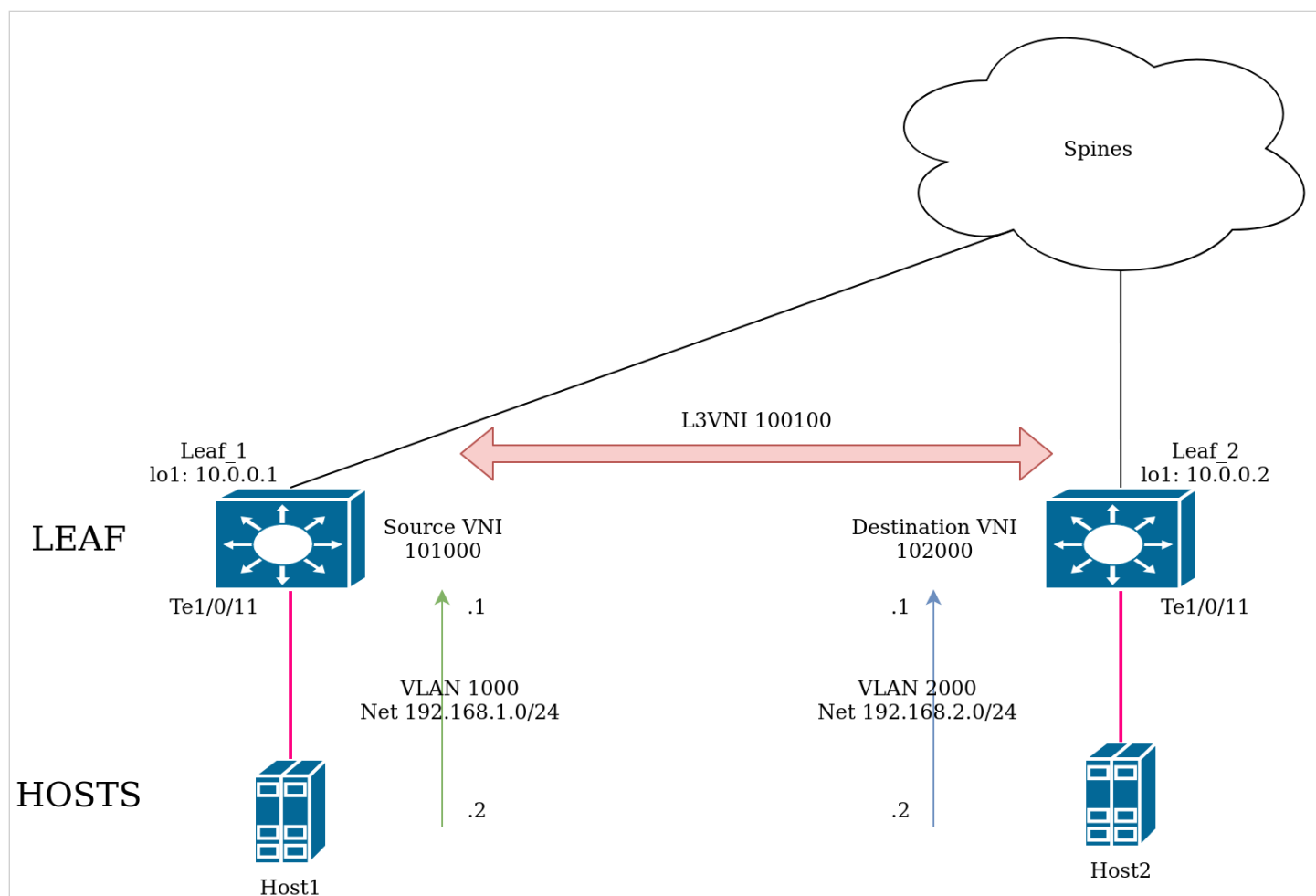
Данная настройка запускает перераспределение маршрутов до connected-сетей в созданный L3VPN. В данном случае организуемый в vrf VRF1.

-  Также возможно перераспределение маршрутов в L3VPN из протокола OSPF и от eBGP-соседей. Для этого необходимо создать и запустить отдельный экземпляр протокола OSPF в соответствующей VRF, который будет работать на стыке IP-фабрики и подключаемой к ней сети. В случае с eBGP достаточно установить сессию с BGP-соседом в соответствующей VRF. Примеры конфигурации Leaf для случаев OSPF и eBGP приведены в [приложении 1](#), в разделе symmetric IRB.

Настроенное перераспределение необходимо для отправки маршрутов типа 5, объявляющих доступность клиентских сетей в данной VRF.

Остальные Leaf в схеме настраиваются аналогично.

Настраиваемая схема маршрутизации представлена ниже.



На устройстве Leaf\_1 создадим VXLAN с VNI 101000 и VLAN 1000. Необходимо добавить IP-интерфейс VLAN 1000 в VRF1, чтобы для маршрутизации пакетов клиента использовалась L3VNI. Затем назначить на этот IP-интерфейс IP-адрес.

```
Leaf_1(config)#vlan database
Leaf_1(config-vlan)#vlan 1000
Leaf_1(config-vlan)#exit
Leaf_1(config)#vxlan test_vxlan1
Leaf_1(config-vxlan)#vni 101000
Leaf_1(config-vxlan)#vlan 1000
Leaf_1(config-vxlan)#exit
Leaf_1(config)#interface vlan 1000
Leaf_1(config-if)#ip vrf VRF1
Leaf_1(config-if)#ip address 192.168.1.1 /24
Leaf_1(config-if)#exit
Leaf_1(config)#interface TenGigabitEthernet1/0/11
Leaf_1(config-if)#description Host1
Leaf_1(config-if)#switchport access vlan 1000
Leaf_1(config-if)#exit
```

Создадим на устройстве Leaf\_2 новую VXLAN с VNI 102000 и VLAN 2000. Аналогично необходимо разместить IP-интерфейс VLAN 2000 в VRF1.

```
Leaf_2(config)#vlan database
Leaf_2(config-vlan)#vlan 2000
Leaf_2(config-vlan)#exit
```

```
Leaf_2(config)#vxlan test_vxlan2
Leaf_2(config-vxlan)#vni 102000
Leaf_2(config-vxlan)#vlan 2000
Leaf_2(config-vxlan)#exit
Leaf_2(config)#interface vlan 2000
Leaf_2(config-if)#ip vrf VRF1
Leaf_2(config-if)#ip address 192.168.2.1 /24
Leaf_2(config-if)#exit
Leaf_2(config)#interface TenGigabitEthernet1/0/11
Leaf_2(config-if)#description Host2
Leaf_2(config-if)#switchport access vlan 2000
Leaf_2(config-if)#exit
```

## 12.2 Проверка настройки

Для проверки работоспособности схемы достаточно проверить L3-связность между хостами. Например, достаточно выполнить команду ping с Host1 до Host2, при условии, что на хостах есть маршруты до сетей друг друга.

Трафик, отправляемый между клиентскими сетями, можно наблюдать при помощи зеркалирования на Spine\_1 (на обоих Spine в случае балансировки неоднородного трафика) инкапсулированным в VXLAN, использующую настроенный L3VNI 100100.

### Зеркалирование трафика

Зеркалирование трафика можно использовать как инструмент контроля на любом устройстве схемы и на любом этапе. Но в случае с инкапсулированным в VXLAN трафиком необходимо руководствоваться следующей информацией: исходящий трафик перехватывается зеркалом до инкапсуляции, поэтому трафик, упакованный в VXLAN, необходимо зеркалировать только на устройстве, для которого он является входящим.

Посмотреть маршрутную информацию в интересующей VRF можно следующей командой: **show ip route vrf WORD<1-32>**.

Пример выполнения команды:

```
Leaf_1#show ip route vrf VRF1
Maximum Parallel Paths: 32 (32 after reset)
Load balancing: src-dst-mac-ip
IP Forwarding: enabled
Codes: > - best, C - connected, S - static,
       R - RIP,
       O - OSPF intra-area, OIA - OSPF inter-area,
       OE1 - OSPF external 1, OE2 - OSPF external 2,
       B - BGP, i - IS-IS, L1 - IS-IS level-1,
       L2 - IS-IS level-2, ia - IS-IS inter area
[d/m]: d - route's distance, m - route's metric

C    192.168.1.0/24 is directly connected, vlan 1000
B    192.168.2.0/24 [200/0] via 10.0.0.2, 00:04:26, VNI 100100,
router-mac e0:d9:e3:17:6b:40
B    192.168.2.2/32 [200/0] via 10.0.0.2, 00:04:26, VNI 100100,
router-mac e0:d9:e3:17:6b:40
```

**⚠** Маршрут до сети 192.168.2.2/32, т.е. до Host2 был получен посредством анонса типа 2 IP-MAC. Этот анонс был отправлен с Leaf\_2 в момент изучения ARP о Host2. Данный маршрут необходим для маршрутизации трафика до конкретного хоста в том случае, если клиентская сеть располагается за несколькими Leaf.

В маршруте до сети 192.168.2.0/24 указан router-mac. Это MAC-адрес роутера назначения. В нашем случае это Leaf\_2.

Данный маршрут получен посредством анонса типа 5.

Маршруты типа 5 можем видеть в выводе маршрутной информации протокола BGP. В примере ниже укороченный вариант вывода, содержащий только маршруты типа 5:

```
Leaf_1#show ip bgp l2vpn evpn

BGP routing table information for VRF default
BGP table version is 10, local router ID is 10.0.0.1
Status codes: * - valid, > - best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Nexthop          Metric      LocPrf      Weight      Path
   .
   .
   .
Route distinguisher: 10.0.0.1:34564
*> [5][0][0][24][192.168.1.0]/224
                0.0.0.0          0           100         0           ?

Route distinguisher: 10.0.0.2:34564
*>i[5][0][0][24][192.168.2.0]/224
                10.0.0.2          0           100         0           ?

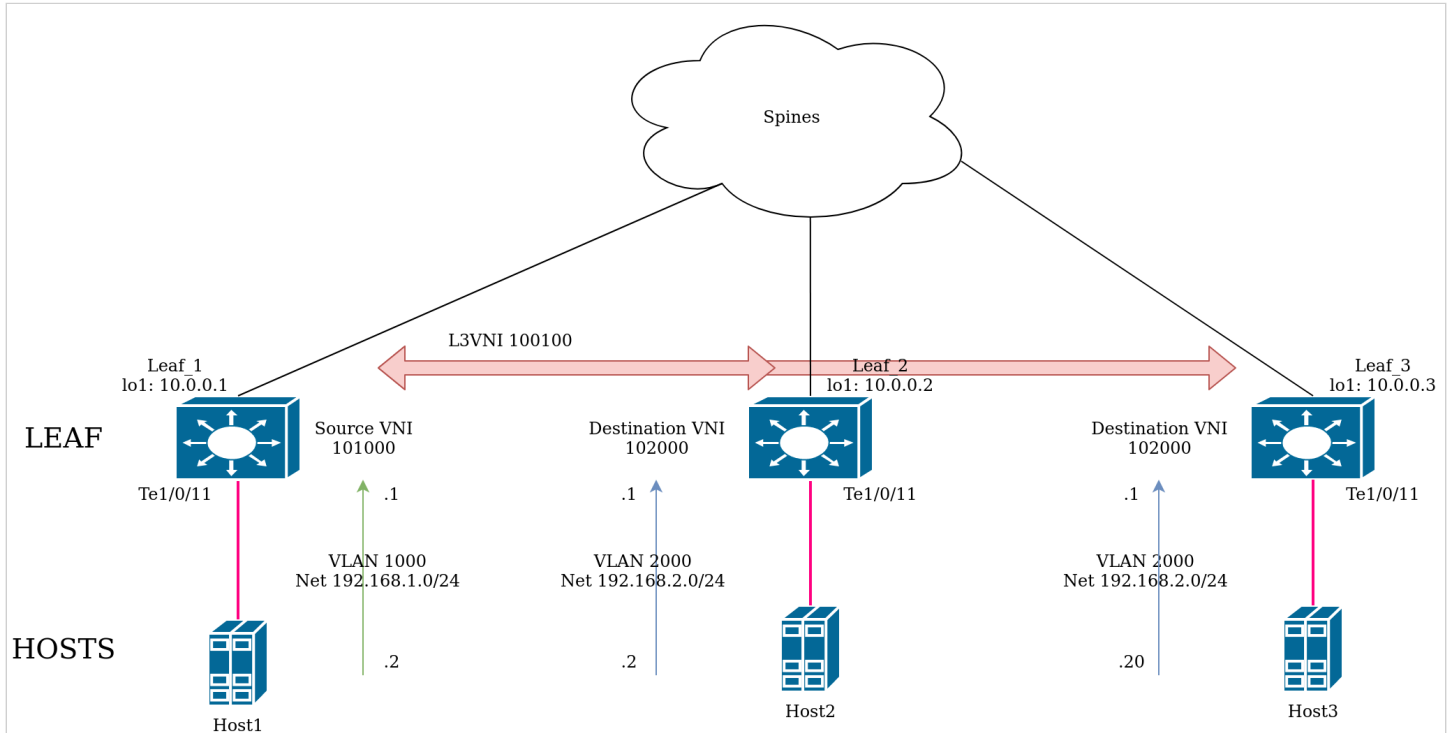
Route distinguisher: 10.0.0.2:34564
* i[5][0][0][24][192.168.2.0]/224
                10.0.0.2          0           100         0           ?
```

Где:

- [5] — тип маршрута;
- [0] — ESI (Ethernet segment identifier);
- [0] — EthTag. В текущей версии ПО не используется;
- [24] — длина префикса;
- [192.168.2.0] — IP-адрес сети назначения;
- 224 — полная длина маршрута.

## 12.3 Anycast gateway

Дополним логическую схему стенда, как показано ниже.



Anycast-gateway дополняет возможность различным Leaf выступать в роли шлюза для одной и той же сети. Хостам при переключении с одного Leaf на другой (например при миграции виртуальных машин) при использовании anycast-gateway нет необходимости обновления ARP-записи, т.к. MAC-адрес шлюза благодаря настройке не меняется от Leaf к Leaf. Подходит как для схем с multihomed-подключением, так и для случаев, когда одна и та же сеть разбита на несколько физических сегментов, подключенных к разным Leaf.

Донастроим Leaf\_3. VXLAN на Leaf\_3, полностью аналогична Leaf\_2, поэтому используем тот же блок настроек. Сеть та же, 192.168.2.0/24.

```
Leaf_3(config)#vlan database
Leaf_3(config-vlan)#vlan 2000
Leaf_3(config-vlan)#exit
Leaf_3(config)#vxlان test_vxlan2
Leaf_3(config-vxlan)#vni 102000
Leaf_3(config-vxlan)#vlan 2000
Leaf_3(config-vxlan)#exit
Leaf_3(config)#interface vlan 2000
Leaf_3(config-if)#ip vrf VRF1
Leaf_3(config-if)#ip address 192.168.2.1 /24
Leaf_3(config-if)#exit
Leaf_3(config)#interface TenGigabitEthernet1/0/11
Leaf_3(config-if)#description Host3
Leaf_3(config-if)#switchport access vlan 2000
Leaf_3(config-if)#exit
```

Настройка anycast-gateway состоит из двух частей.

1. Задать виртуальный MAC-адрес, который заменит базовый MAC-адрес коммутатора в ARP-пакетах, исходящих с интерфейсов, на которых задействована данная функция. Пример для Leaf\_2 и VLAN 2000:

```
Leaf_2(config)#anycast-gateway mac-address 00:00:00:11:11:11
```

2. Включить anycast gateway на VLAN-интерфейсе:

```
Leaf_2(config)#interface vlan 2000
Leaf_2(config-if)#anycast-gateway
```

3. Аналогичным образом настроить Leaf\_3, используя тот же виртуальный MAC-адрес.

Посмотреть выполненные настройки можно show-командой:

```
Leaf_2#show ip anycast-gateway
Anycast-gateway virtual MAC address: 00:00:00:11:11:11

Anycast-gateway is configured on interfaces:

Vlans:
  2000
```

### Проверка настройки anycast-gateway

1. Выполнить ping между всеми хостами. В результате: есть L3 связность между Host1 и двумя другими хостами; есть L2 связность между Host2 и Host3;
2. В ARP-таблице хостов 2 и 3 присутствует соответствие настроенного адреса шлюза и виртуального MAC-адреса anycast-gateway. Пример просмотра таблицы на Host3:

```
Host3#sh arp

Total number of entries: 1

-----
```

VLAN	Interface	IP address	HW address	status	IP Unnumbered I/F
vlan 1	te1/0/1	192.168.2.1	00:00:00:11:11:11	dynamic	

```
-----
```

## 12.4 Gateway-ip для маршрутов типа 5

Данная функция позволяет установить значение поля IPv4 Gateway address в NLRI отправляемых EVPN-маршрутов типа 5.

Установка происходит посредством привязки к соседу route-map с соответствующим действием set.

**⚠** Route-map с данной настройкой следует использовать в контексте address-family l2vpn evpn BGP-соседа и только в направлении out.

Т.к. в случае IP-фабрики соседями для Leaf являются Spine-коммутаторы, для корректной работы фабрики необходимо настраивать данную route-map в сторону всех Spine.

Пример настройки для Leaf\_1:

```
Leaf_1(config)#route-map TESTMAP 10 permit
Leaf_1(config-route-map)#set evpn gateway-ip 192.168.50.1
Leaf_1(config-route-map)#exit
Leaf_1(config)#router bgp 65500
Leaf_1(router-bgp)#neighbor 10.0.1.1
Leaf_1(router-bgp-nbr)#address-family l2vpn evpn
Leaf_1(router-bgp-nbr-af)#route-map TESTMAP out
Leaf_1(router-bgp-nbr-af)#exit
Leaf_1(router-bgp-nbr)#exit
Leaf_1(router-bgp)#neighbor 10.0.2.2
Leaf_1(router-bgp-nbr)#address-family l2vpn evpn
Leaf_1(router-bgp-nbr-af)#route-map TESTMAP out
Leaf_1(router-bgp-nbr-af)#exit
Leaf_1(router-bgp-nbr)#exit
```

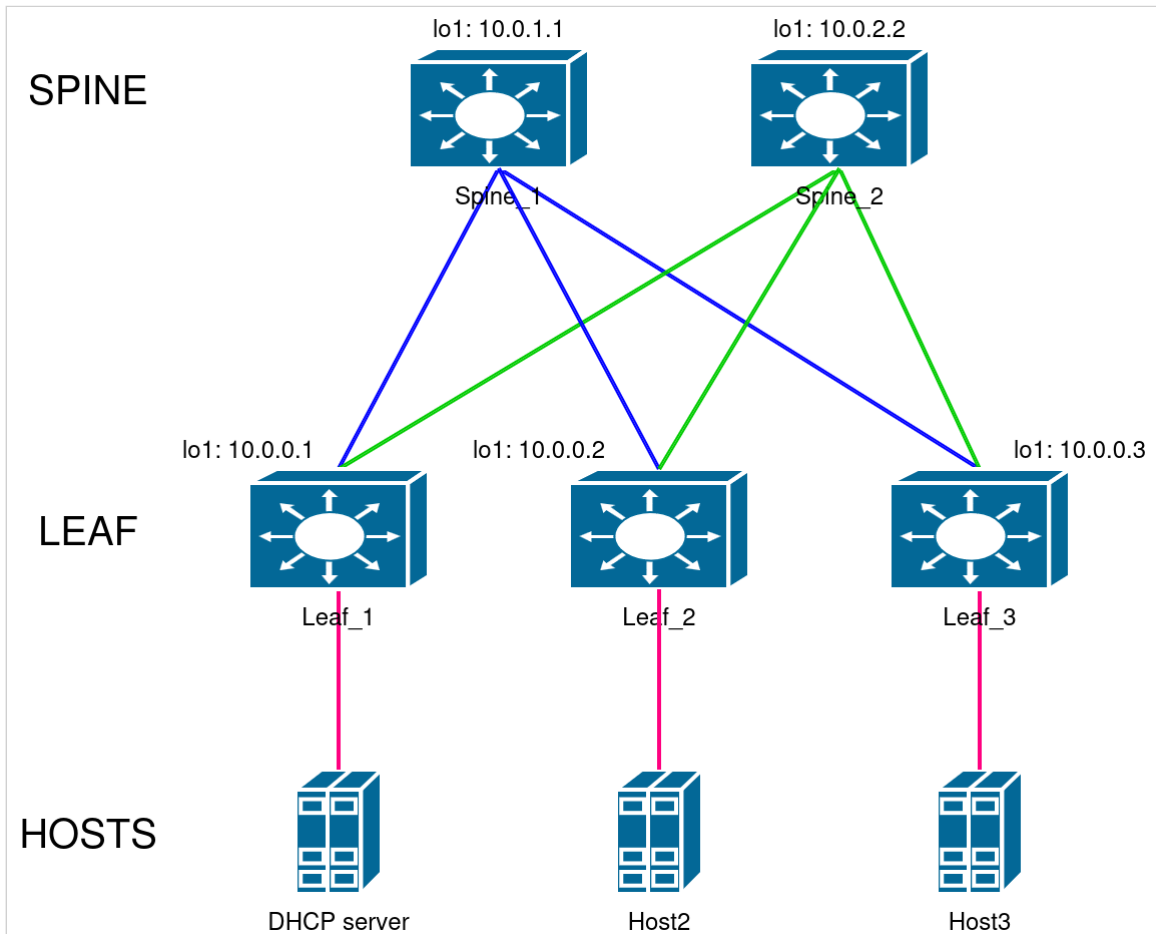
Результат можем наблюдать в поле IPv4 Gateway address в NLRI отправляемых маршрутов типа 5:

```
EVPN NLRI: IP Prefix route
Route Type: IP Prefix route (5)
Length: 34
Route Distinguisher: 00010a0000018704 (10.0.0.1:34564)
ESI: 00:00:00:00:00:00:00:00:00
Ethernet Tag ID: 0
IP prefix length: 24
IPv4 address: 192.168.1.0
IPv4 Gateway address: 192.168.50.1
```

Удаленные Leaf будут устанавливать полученные маршруты в таблицы маршрутизации с учетом указанного значения IPv4 Gateway address и доступности указанного шлюза.

## 12.5 DHCP relay

Используемая схема представлена ниже.



В качестве отправных конфигураций устройств в данном разделе используются конфигурации из приложения [Конфигурации для Symmetric IRB](#).

Дополнительные настройки, необходимые для работы агента DHCP relay, описаны далее в данном разделе.

### Настройка DHCP relay

На Leaf\_1 со стороны сервера DHCP дополнительных настроек не требуется. Разместим сервер в существующей VLAN 1000, интерфейс TenGigabitEthernet1/0/11.

**⚠** Сценарий 1. Host2 и Host3 находятся в одной сети, но на разных Leaf. DHCP-сервер находится в другой сети. Соответственно необходимо решить задачу отправки ответа от сервера DHCP на нужный Leaf, при этом выдав IP-адрес хосту из нужного пула адресов.

В примере ниже настроим Leaf\_2.

Первым делом необходимо создать дополнительный loopback интерфейс в VRF "VRF1", который будет являться интерфейсом-источником для DHCP-запросов, переадресуемых на DHCP-сервер. Адреса этих loopback также должны быть уникальны для каждого Leaf. Именно это позволит ответам от DHCP-сервера возвращаться на нужный Leaf.

```
Leaf_2(config)#interface loopback2
Leaf_2(config-if)#ip vrf VRF1
Leaf_2(config-if)#ip address 10.0.0.102 255.255.255.255
Leaf_2(config-if)#exit
```

На DHCP-сервере необходимо наличие маршрутов до этих loopback-интерфейсов или наличие маршрута по умолчанию для возможности отправки ответов на DHCP-запросы.

Включим функцию DHCP relay-агента на VLAN, в которую включен Host2. В нашем случае VLAN 2000:

```
Leaf_2(config)#interface vlan 2000
Leaf_2(config-if)#ip dhcp relay enable
Leaf_2(config-if)#exit
```

Теперь выполним все необходимые настройки в режиме глобальной конфигурации:

```
Leaf_2(config)#ip dhcp relay address 192.168.11.2 vrf VRF1
Leaf_2(config)#ip dhcp relay enable
Leaf_2(config)#ip dhcp relay source-interface loopback 2 vrf VRF1
Leaf_2(config)#ip dhcp information option
Leaf_2(config)#ip dhcp information option vpn link-selection
```

- *ip dhcp relay address 192.168.11.2 vrf VRF1* — задает адрес DHCP-сервера, куда будут перенаправляться DHCP-запросы в пределах VRF. При передаче между Leaf, эти пакеты будут инкапсулированы в VXLAN с использованием L3 VNI. В нашем случае VNI 100100;
- *ip dhcp relay enable* — включает функцию DHCP relay-агента глобально на устройстве;
- *ip dhcp relay source-interface loopback 2 vrf VRF1* — задает интерфейс-источник для DHCP-запросов, переадресуемых на DHCP-сервер, в пределах VRF. При этом в DHCP-пакетах в сторону сервера будет заменяться src ip и giaddr (Relay agent IP address) на адрес интерфейса loopback2;
- *ip dhcp information option* — включает добавление опции 82 в DHCP-запрос;
- *ip dhcp information option vpn link-selection* — включает добавление подопции 5 в опцию 82 DHCP-запроса. При указании только параметра *vpn* будут добавлены все подопции: 5, 11 и 151.

### Подопция 5

Link Selection. Содержит адрес сети клиента. Позволяет разделять giaddr (Relay agent IP address) и сеть клиента, тем самым давая возможность DHCP-серверу выдавать IP-адреса на основании значения этой подопции.


### Подопция 11

Server ID Override. Содержит IP-адрес интерфейса, на котором включен relay-агент. При включении этой подопции, устройство перезаписывает опцию 54 (DHCP Server Identifier) в ответах от DHCP-сервера, подставляя туда значение из подопции 11.

### Подопция 151

Virtual Subnet Selection. Содержит имя VRF на клиентском интерфейсе. Также может использоваться для выдачи адресов DHCP-сервером из определенного пула.

Leaf\_3 настраивается аналогичным образом за исключением IP-адреса на интерфейсе loopback2.

 Пример настройки `isc-dhcp-server` для обработки подопции 5 в конце данного раздела.

## Ожидаемое поведение

- DHCP запросы от Host2 и 3 будут перехватываться на Leaf DHCP Relay-агентом;
- При отправке в сторону DHCP-сервера в пакеты будет добавлена опция 82 с подопцией 5 с соответствующим содержанием;

```
Option: (82) Agent Information Option
  Length: 38
  Option 82 Suboption: (1) Agent Circuit ID
  Option 82 Suboption: (2) Agent Remote ID
  Option 82 Suboption: (5) Link selection (192.168.2.0)
    Length: 4
    Link selection: 192.168.2.0
```

- В поле Relay agent IP address будет вставлен IP-адрес интерфейса loopback2;
- DHCP-сервер может выдавать IP-адреса из различных пулов, используя подопцию 5 опции 82;
- Ответы от DHCP-сервера направляются на Leaf, с Relay-агента которого поступил запрос, благодаря уникальным адресам интерфейсов Loopback 2, которые используются в качестве Relay agent IP address в DHCP-запросах.

**⚠** Сценарий 2. Host2 и Host3 находятся в разных сетях. Каждая клиентская сеть имеет подключение только к одному коммутатору Leaf. Соответственно IP-адреса всех Relay-агентов будут уникальные.

Интерфейсы в сторону хостов на Leaf выглядят следующим образом:

```
Leaf_2:

interface vlan 2000
 ip vrf VRF1
 ip address 192.168.2.1 255.255.255.0
 ip dhcp relay enable
 anycast-gateway
!

Leaf_3:

interface vlan 2000
 ip vrf VRF1
 ip address 192.168.3.1 255.255.255.0
 ip dhcp relay enable
 anycast-gateway
!
```

При данном сценарии настройку Leaf можно упростить. Для корректной работы DHCP Relay достаточно следующих настроек в режиме глобальной конфигурации:

```
Leaf_2(config)#ip dhcp relay address 192.168.11.2 vrf VRF1
Leaf_2(config)#ip dhcp relay enable
```

Создание дополнительных Loopback-интерфейсов на Leaf не требуется.

На DHCP-сервере необходимо наличие маршрутов до клиентских сетей или наличие маршрута по умолчанию для возможности отправки ответов на DHCP-запросы.

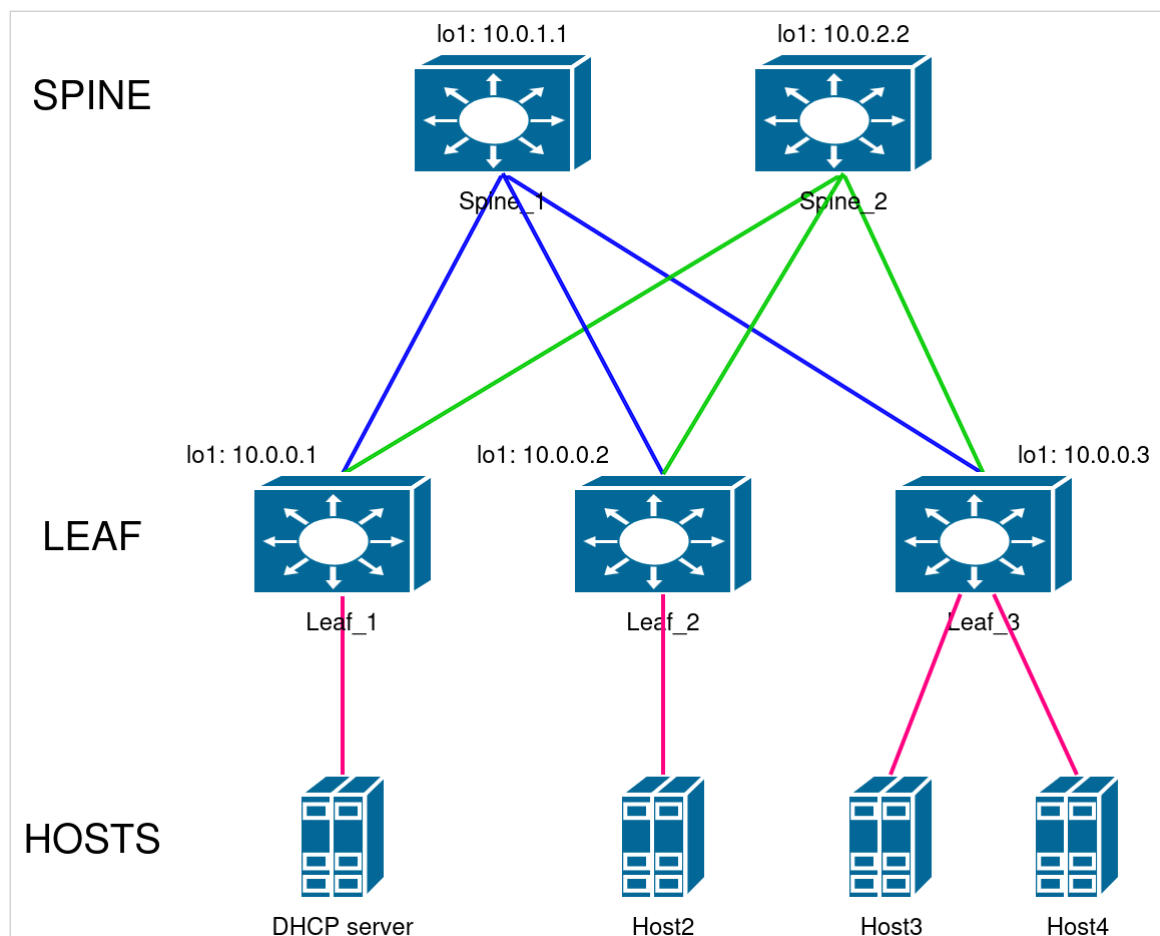
**⚠** Пример настройки isc-dhcp-server для обработки подопции 5 в конце данного раздела.

## Ожидаемое поведение

- DHCP-запросы от Host2 и 3 будут перехватываться на Leaf DHCP Relay агентом;
- При этом в DHCP-пакетах в сторону сервера будет заменяться src ip и giaddr (Relay agent IP address) на адрес, настроенный на клиентских интерфейсах, откуда поступил DHCP-запрос. Для Leaf\_2 это адрес 192.168.2.1, для Leaf\_3 это 192.168.3.1;
- DHCP-сервер может выдавать IP-адреса из различных пулов согласно значению поля Relay agent IP address в DHCP-запросах.

⚠ Сценарий 3. Host2 и Host3 находятся в одной сети, но на разных Leaf. Host4 находится в другой сети.

## Схема



Для Host4 создан свой VXLAN. Интерфейс для сети Host4 также должен находиться в VRF1:

```
vxlan test_vxlan4
 vni 102004
  vlan 2004
exit
!
interface vlan 2004
 ip vrf VRF1
 ip address 192.168.4.1 255.255.255.0
 ip dhcp relay enable
exit
!
```

Решением данной задачи будет являться настройка по сценарию 1.

Благодаря уникальным адресам интерфейсов Loopback 2, ответы от DHCP-сервера будут направляться на тот Leaf, с Relay-агента которого поступил запрос. А благодаря подопции 5 опции 82 будет возможность выдавать адреса из определенного пула.

### Пример настройки isc-dhcp-server для обработки подопции 5

**⚠** Данный пример приведен для случая, когда версия isc-dhcp-server не поддерживает автоматическое распознавание подопции 5 опции 82. В противном случае нет необходимости в настройках "class" и "allow members".

```
class "192.168.2.0" {
    match if binary-to-ascii (10, 8, ".", option agent.link-selection) = "192.168.2.0";
}

class "192.168.3.0" {
    match if binary-to-ascii (10, 8, ".", option agent.link-selection) = "192.168.3.0";
}

class "192.168.4.0" {
    match if binary-to-ascii (10, 8, ".", option agent.link-selection) = "192.168.4.0";
}

#-----For Leaf_2-----
subnet 192.168.2.0 netmask 255.255.255.0 {
    pool {
        allow members of "192.168.2.0";
        range 192.168.2.10 192.168.2.254;
        option subnet-mask 255.255.255.0;
        option routers 192.168.2.1;
    }
}

#-----For Leaf_3-----
subnet 192.168.3.0 netmask 255.255.255.0 {
    pool {
        allow members of "192.168.3.0";
        range 192.168.3.10 192.168.3.254;
        option subnet-mask 255.255.255.0;
        option routers 192.168.3.1;
    }
}

#-----For Leaf_3_2-----
subnet 192.168.4.0 netmask 255.255.255.0 {
    pool {
        allow members of "192.168.4.0";
        range 192.168.4.10 192.168.4.254;
        option subnet-mask 255.255.255.0;
        option routers 192.168.4.1;
    }
}
```

Классы позволяют осуществлять выборку по значению подопции 5 опции 82.

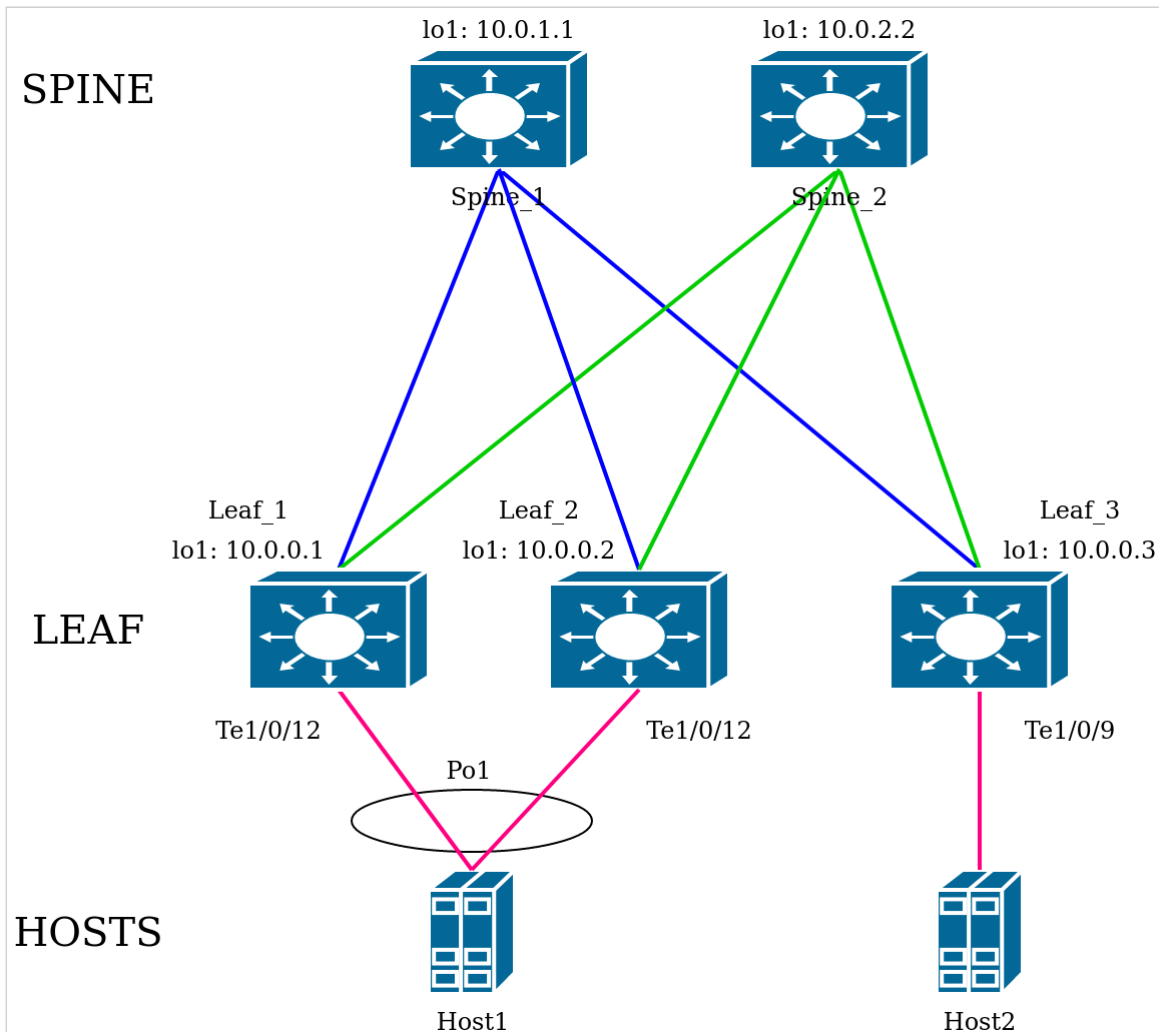
Условием выдачи адреса из определенного пула является принадлежность запроса определенному классу.

## 12.6 Ожидаемый результат

- L3VNI обеспечивает симметричную маршрутизацию между клиентскими сетями, размещенными на разных Leaf и в разных VXLAN;
- Функция anycast-gateway позволяет подменять MAC-адрес шлюза, создаваемого на устройствах Leaf;
- Есть возможность установить значение поля IP-v4 Gateway address в EVPN маршрутах типа 5;
- Функция DHCP Relay-агента позволяет перенаправлять DHCP-пакеты на сервер и обратно, добавляя к запросу необходимые опции.

## 13 EVPN multihoming

Используемая схема представлена ниже.



В качестве отправных конфигураций устройств в данном разделе используются конфигурации из приложения [Конфигурации для multicast VXLAN](#).

- ⚠️ Работу EVPN multihoming в данном разделе рассматриваем на примере multicast VXLAN, поскольку такой вариант является более ёмким в плане настроек. Также можно использовать и режим ingress replication. В таком случае настройки в части multicast не требуются.

Дополнительные настройки, необходимые для работы EVPN multihoming, описаны далее в данном разделе.

- ⚠️ Для уменьшения времени сходимости топологии при multihomed-подключении к IP-фабрике рекомендуется использовать уменьшенные интервалы между отправкой update-сообщений протокола BGP на всех Leaf-коммутаторах. Настраиваются они в контексте конфигурации BGP-соседа.

Пример настройки BGP для Leaf\_1:

```
Leaf_1(config)#router bgp 65500
Leaf_1(router-bgp)#neighbor 10.0.1.1
Leaf_1(router-bgp-nbr)#advertisement-interval 1 withdraw 1
Leaf_1(router-bgp-nbr)#exit
Leaf_1(router-bgp)#neighbor 10.0.2.2
Leaf_1(router-bgp-nbr)#advertisement-interval 1 withdraw 1
```

В [приложении 1](#) есть полные конфигурации устройств, получаемые по окончании данного раздела.

**⚠** Работа EVPN/VXLAN в связке с MLAG (VPC) не поддерживается в текущей версии ПО.

### 13.1 Настройка EVPN multihoming

Host1 будет выступать в роли multihomed-клиента. Соединение с парой Leaf\_1-2 осуществляется посредством port-channel с использованием LACP.

Выполним донастройку Leaf\_1 и Leaf\_2:

```
Leaf_1(config)#interface port-channel 1
Leaf_1(config-if)#description Host1_LAG
Leaf_1(config-if)#switchport mode trunk
Leaf_1(config-if)#switchport trunk allowed vlan add 2-5
Leaf_1(config-if)#hold-time up 60
Leaf_1(config-if)#ethernet-segment 1000
Leaf_1(config-es)#system-mac 11:22:33:44:55:66
Leaf_1(config-es)#exit
Leaf_1(config-if)#interface TenGigabitEthernet1/0/12
Leaf_1(config-if)#channel-group 1 mode auto
Leaf_1(config-if)#exit
```

Где:

- hold-time up 60 – устанавливает задержку добавления порта в port-channel. Величина задержки выбирается исходя из размеров ЦОД и времени сходимости протокола BGP в конкретных условиях. Значения варьируются в диапазоне 0-300 секунд;
- ethernet-segment 1000 – создает Ethernet-сегмент с номером 1000;
- system-mac 11:22:33:44:55:66 – задает MAC-адрес, используемый в качестве System ID протокола LACP.

Настройки Leaf\_2 идентичны.

Следующие функции работают по умолчанию и не требуют дополнительных настроек:

- Фильтрация Split horizon BUM-трафика – исключает возможность возвращения BUM-трафика, вышедшего из Ethernet-сегмента в этот же сегмент через другие Leaf, имеющие соединение с этим сегментом;
- Designated forwarder – только устройство, выбранное DF, обладает правом отправлять BUM-трафик в Ethernet-сегмент;
- Local bias – если Ethernet-сегмент источника и Ethernet-сегмент назначения имеют соединение с одним и тем же Leaf, то трафик между ними не отправляется в сторону Spine;
- Fast convergence – VTEP отправляет сообщение BGP withdraw типа 1 при потере линка в Ethernet-сегменте. Механизм fast convergence позволяет удаленным VTEP удалить MAC-адреса, изученные на этом VTEP/ES при получении BGP withdraw типа 1.

## 13.2 Core isolation

Представим ситуацию, что при multihomed-подключении к IP-фабрике возникла авария, при которой коммутатор Leaf лишился всех uplink-соединений до Spine. Multihomed-клиент продолжит балансировать отправляемый вверх трафик во все линки LAG. Следовательно, тот трафик, который пойдет на аварийный Leaf, будет потерян. Чтобы этого избежать, разработан механизм, позволяющий коммутатору Leaf, потерявшему все uplink-соединения, отключить на время аварии LAG, участвующие в ES.

Включается данный механизм в контексте BGP-соседа и ориентируется на состояние соседства. Пример настройки для Leaf\_1:

```
Leaf_1(config)#router bgp
Leaf_1(router-bgp)#neighbor 10.0.1.1
Leaf_1(router-bgp-nbr)#evpn multihoming core-tracking
Leaf_1(router-bgp-nbr)#exit
Leaf_1(router-bgp)#neighbor 10.0.2.2
Leaf_1(router-bgp-nbr)#evpn multihoming core-tracking
```

Как только последнее соседство, на котором задействован multihoming core-tracking, перейдет в состояние down, устройство отключит все порты, участвующие во всех ethernet-segment.

Пример log-сообщений при срабатывании:

```
%BGP-I-ADJCHANGE: Neighbor 10.0.2.2 Down
%LINK-W-Down: te1/0/12
%2SWTRUNK-I-TRNKPORTPARAM: auto-negotiation/adv. capabilities of port te1/0/12 differ from auto-
negotiation/adv. capabilities of Po1
%LINK-W-PORT_SUSPENDED: Port Po1 suspended by evpn-core-tracking
```

Восстановление работы выключенных портов происходит при установлении хотя бы одного соседства-трекера.

Есть возможность задать таймер для отложенного восстановления портов-участников ethernet-segment:

```
Leaf_1(config)#evpn ethernet-segment multihoming core-tracking delay time
<1-300> Delay time in seconds
```

Значение по умолчанию 0, т.е. задержка восстановления отсутствует.

Для просмотра оперативной информации используется show-команда:

```
Leaf_1#show evpn ethernet-segment multihoming core-tracking

EVPN Core status is CONNECTED

Multihoming ES UP delay timer is disabled

Peer trackings:

  BGP neighbor      State
  -----
  10.0.1.1          ESTABLISHED
  10.0.2.2          IDLE

Ethernet segments:
```

Interface	Status admin/oper	Ethernet Segment ID (ESI)
Po1	UP/UP	03:11:22:33:44:55:66:00:03:e8

В примере вывода отсутствует соседство со вторым Spine и не задан таймер отложенного восстановления.

### 13.3 Проверка настройки

**⚠** Поскольку интерфейсы Po1 на коммутаторах Leaf настроены в режиме trunk, интерфейсы Host1 и Host2 также должны быть настроены в режиме trunk с указанием VLAN 2.

Со стороны Host1 необходимо проверить, что оба соединения с Leaf были добавлены в port-channel.

На Leaf аналогично можно проверить состояние настроенного port-channel:

```
Leaf_1#show interfaces Port-Channel1
Port-Channel1 is up (connected)
  Interface index is 1000
  Hardware is aggregated ethernet interface(s), MAC address is cc:9d:a2:53:d6:81
  Description: Host1_LAG
  Interface MTU is 9000
  Link is up for 0 days, 0 hours, 20 minutes and 13 seconds
  Link aggregation type is LACP
  No. of members in this port-channel: 1 (active 1)
    TenGigabitEthernet1/0/12, full-duplex, LACP active, 10000Mbps (active)
  Active bandwidth is 10000Mbps
  15 second input rate is 0 Kbit/s
  15 second output rate is 0 Kbit/s
    83 packets input, 7936 bytes received
    0 broadcasts, 83 multicasts
    0 input errors, 0 FCS
    0 oversize, 0 internal MAC
    0 pause frames received
  184 packets output, 45824 bytes sent
  101 broadcasts, 83 multicasts
  0 output errors, 0 collisions
  N/S excessive collisions, 0 late collisions
  0 pause frames transmitted
```

Проверим теперь состояние ethernet-segment:

```
Leaf_1#show evpn ethernet-segment

Interface Status      ES Number   Ethernet Segment ID (ESI)   Remote members
  admin/oper
-----
Po1      UP/UP       1000       03:11:22:33:44:55:66:00:03:e8   10.0.0.2
```

```
Leaf_1#show evpn ethernet-segment 03:11:22:33:44:55:66:00:03:e8 detailed
```

```
Ethernet Segment: Po1
  ESI: 03:11:22:33:44:55:66:00:03:e8
  ES number is 1000
  ES system MAC address is 11:22:33:44:55:66
  Administrative status is up
  Operational status is up
```

```
All-Active multi-homing mode
Route Distinguisher is 10.0.0.1:0 (auto-assigned)
Route Target is 1122.3344.5566 (auto-assigned)
Designated Forwarder election delay is 3 seconds
Members:
```

Router ID	Type
10.0.0.1	local
10.0.0.2	remote

Connected VXLANs:

VNI	VLAN ID	Designated Forwarder	Name
102	2	10.0.0.1	mcast2
103	3	10.0.0.2	mcast3
104	4	10.0.0.1	mcast4
105	5	10.0.0.2	mcast5

В выводе детальной информации можно увидеть настроенные ES number и ES system MAC address, а также сгенерированный на их основе ESI: 03:11:22:33:44:55:66:00:03:e8.

Кроме того, можно видеть информацию обо всех устройствах-членах Ethernet-сегмента. В данном случае о Router ID 10.0.0.1 и 10.0.0.2.

Выбор Designated Forwarder может отличаться для различных VXLAN, что можно видеть в таблице Connected VXLANs. Именно DF отвечает за отправку BUM-трафика в конкретный Ethernet-сегмент в пределах конкретной VLAN.

В таблице MAC-адресов Leaf\_3 должен присутствовать MAC-адрес Host1 (вывод информации сокращен для наглядности примера):

```
Leaf_3#show mac address-table
Flags: I - Internal usage VLAN
Aging time is 300 sec
```

Vlan	Mac Address	Interface	Type
2	e0:d9:e3:a8:45:40	10.0.0.2 10.0.0.1	evpn-vxlan

Где:

- e0:d9:e3:a8:45:40 – MAC-адрес Host1. В поле интерфейс указаны IP-адреса Leaf\_1 и Leaf\_2, что говорит о доступности этого MAC-адреса через два туннеля.

В выводе маршрутной информации BGP будут маршруты типа 2 с указанным ESI (вывод информации сокращен для наглядности примера):

```
Leaf_3#show ip bgp l2vpn evpn

BGP routing table information for VRF default
BGP table version is 60, local router ID is 10.0.0.3
Status codes: * - valid, > - best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Nexthop	Metric	LocPrf	Weight	Path
---------	---------	--------	--------	--------	------

```

Route distinguisher: 10.0.0.2:2
*>i [2] [03:11:22:33:44:55:66:00:03:e8] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216
      10.0.0.2          0          100          0          ?

Route distinguisher: 10.0.0.2:2
* i [2] [03:11:22:33:44:55:66:00:03:e8] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216
      10.0.0.2          0          100          0          ?

```

Где:

- [2] – тип маршрута;
- [03:11:22:33:44:55:66:00:03:e8] – ESI (Ethernet segment identifier). Создается на основе настроенных номера Ethernet-сегмента и system mac;
- [0] – EthTag. В текущей версии ПО не используется;
- [48] – длина MAC-адреса;
- [e0:d9:e3:a8:45:40] – MAC-адрес Host1, изученный на удаленном VTEP;
- [0] – длина IP-адреса;
- [0.0.0.0] – IP-адрес;
- 216 – полная длина маршрута.

В качестве проверки работоспособности схемы с multihoming подключением можно использовать проверку IP-связности между Host1 и Host2. Для этого их IP-интерфейсы во VLAN 2 должны быть в одной подсети.

Поочередное отключение линков между Host1 и Leaf-ами не приведет к потере IP-связности.

### 13.4 Ожидаемый результат

- Есть возможность подключения клиентского устройства посредством LAG к нескольким Leaf;
- Передача пользовательского трафика между multihomed и singlehomed-клиентами успешна;
- Благодаря различным механизмам фильтрации BUM-трафика не возникает петель при обмене данными с multihomed-клиентом.

## 14 Команды для просмотра информации

В данном разделе собраны show-команды, касающиеся описываемого в этом руководстве функционала. Присутствуют описания и примеры выводов.

- **show vxlan [word]**

Отображает краткую информацию по всем созданным VXLAN. Есть возможность фильтрации вывода по имени конкретной VXLAN с отображением более детализированной информации.

Пример вывода:

```
Leaf_1#show vxlan
```

Name	VNI	VLAN ID	Status admin/oper	BUM Forwarding	Route Distinguisher
L3_vxlan	100100	100	UP/UP	Ingress Repl.	10.0.0.1:100
test_vxlan	101000	1000	UP/UP	Ingress Repl.	10.0.0.1:1000
test_vxlan1	101001	1001	UP/UP	Ingress Repl.	10.0.0.1:1001
test_vxlan2	101002	1002	UP/UP	Ingress Repl.	10.0.0.1:1002

```
Leaf_1#show vxlan test_vxlan
```

```
test_vxlan
VxLAN Network ID is 101000, VLAN ID is 1000
Administrative status is UP
Operational status is UP

Local Router ID is 10.0.0.1
Route Distinguisher is 10.0.0.1:1000 (auto-assigned)
Route Target is: 65535:268536456 (auto-assigned)

Broadcast/Unknown Unicast/Multicast traffic
is forwarded in Ingress Replication mode
```

- **show vxlan tunnels [word]**

Отображает информацию обо всех установленных VXLAN-туннелях. Есть возможность фильтрации вывода по имени конкретной VXLAN.

Пример вывода:

```
Leaf_1#show vxlan tunnels
```

```
test_vxlan
```

Destination	Source	Status
101000:10.0.0.2	101000:10.0.0.1	Up
101000:10.0.0.3	101000:10.0.0.1	Up

```
test_vxlan1
```

Destination	Source	Status
101001:10.0.0.2	101001:10.0.0.1	Up
101001:10.0.0.3	101001:10.0.0.1	Up

```
test_vxlan2
```

Destination	Source	Status
-------------	--------	--------

```

-----
101002:10.0.0.2          101002:10.0.0.1      Up
101002:10.0.0.3          101002:10.0.0.1      Up

Leaf_1#show vxlan tunnels test_vxlan

test_vxlan

Destination              Source                  Status
-----
101000:10.0.0.2          101000:10.0.0.1      Up
101000:10.0.0.3          101000:10.0.0.1      Up

```

- **show arp suppression-cache [local | remote | vlan]**

Отображает содержимое кэша arp suppression. Есть возможность фильтрации вывода по типам записей и VLAN.

Пример вывода:

```

Leaf_1#show arp suppression-cache

Total number of entries: 2
ARP suppression-cache timeout is 300 sec

IP address      Vtep      MAC address      VLAN      Port      Flags      Age
-----
192.168.13.1    0.0.0.0   e0:d9:e3:d7:ea:80  1000      te1/0/11  local      00:00:33
192.168.13.2    10.0.0.3  e0:d9:e3:26:d6:00  1000      0         remote     00:00:08

```

- **show ip anycast-gateway**

Выводит информацию об anycast-gateway. Здесь можно увидеть настроенный виртуальный MAC-адрес и интерфейсы, на которых активирована данная функция.

Пример вывода:

```

Leaf_2#show ip anycast-gateway
Anycast-gateway virtual MAC address: 00:00:00:11:11:11

Anycast-gateway is configured on interfaces:

Vlans:
  2000

```

- **sh mac address-table [count]**

Данная команда отображает таблицу изученных MAC-адресов. MAC-адреса, изученные посредством EVPN, имеют тип evpn-vxlan.

Пример вывода:

```

Leaf_1#show mac address-table
Flags: I - Internal usage VLAN
Aging time is 300 sec

Vlan      Mac Address      Interface      Type
-----
  1        e0:d9:e3:26:d6:00  0              self

```

1000	0c:9d:92:61:9f:c4	te1/0/11	dynamic
1000	e0:d9:e3:a8:45:40	10.0.0.3	evpn-vxlan
1000	e0:d9:e3:a8:55:40	10.0.0.3	evpn-vxlan
1000	e0:d9:e3:a8:65:40	10.0.0.3	evpn-vxlan
hu1/0/1(I)	cc:9d:a2:53:d6:80	hu1/0/1	dynamic
hu1/0/1(I)	cc:9d:a2:53:d6:81	hu1/0/1	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:40	hu1/0/2	dynamic
hu1/0/2(I)	e0:d9:e3:17:6b:41	hu1/0/2	dynamic

Если к команде добавить модификатор вывода **count**, то там можно увидеть счетчик Service dynamic:

```
Leaf_1#show mac address-table count
Capacity      : 131072
Free          : 131062
Used unicast  : 8
Used multicast : 0
Used IPv4 hosts : 2
Used IPv6 hosts : 0 (each IPv6 host consumes 2 entries in MAC address table)
Secure        : 0
Dynamic unicast : 3
Static unicast : 0
Internal      : 5
Service dynamic : 3
```

В этом счетчике учитываются MAC-адреса, изученные посредством EVPN.

- **show ip bgp l2vpn evpn**

Данная команда отображает маршрутную информацию протокола BGP. Модификатор вывода l2vpn evpn позволяет отобразить только содержимое, относящееся к AFI/SAFI l2vpn evpn.

Далее рассмотрим типы маршрутов EVPN.

#### Маршрут типа 1 (per ESI):

```
Route distinguisher: 10.0.0.1:0
*>i [1] [03:11:22:33:44:55:66:00:03:e8] [-1]/152
      10.0.0.1      0      100      0      ?
```

Где:

- [1] – тип маршрута;
- [03:11:22:33:44:55:66:00:03:e8] – ESI (Ethernet segment identifier);
- [-1] – EthTag. В текущей версии ПО не используется;
- 152 – полная длина маршрута.

Маршрут данного типа создается один для всех EVPN-инстансов с линком в данном ESI. RD (в примере 10.0.0.1:0) в данном маршруте будет заканчиваться на ноль.

#### Маршрут типа 1 (per EVI):

```
Route distinguisher: 10.0.0.1:1000
*>i [1] [03:11:22:33:44:55:66:00:03:e8] [0]/152
      10.0.0.1      0      100      0      ?
```

Где:

- [1] – тип маршрута;
- [03:11:22:33:44:55:66:00:03:e8] – ESI (Ethernet segment identifier);
- [0] – EthTag. В текущей версии ПО не используется;
- 152 – полная длина маршрута.

Маршрут данного типа создается каждым конкретным экземпляром с линком в данном ESI.

### Маршрут типа 2 (mac):

Network	Nexthop	Metric	LocPrf	Weight	Path
Route distinguisher: 10.0.0.3:1000					
*>i [2] [0] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216					
	10.0.0.3	0	100	0	?

Где:

- [2] – тип маршрута;
- [0] – ESI (Ethernet segment identifier);
- [0] – EthTag. В текущей версии ПО не используется;
- [48] – длина MAC-адреса;
- [e0:d9:e3:a8:45:40] – MAC-адрес, изученный на удаленном VTEP;
- [0] – длина IP-адреса;
- [0.0.0.0] – IP-адрес;
- 216 – полная длина маршрута.

Маршрут данного типа используется для передачи информации об изученных в пределах VXLAN MAC-адресах.

### Маршрут типа 2 (mac-ip):

Route distinguisher: 10.0.0.2:2000					
*>i [2] [0] [0] [48] [e0:d9:e3:17:6b:40] [32] [192.168.2.2] /272					
	10.0.0.2	0	100	0	?

Где:

- [2] – тип маршрута;
- [0] – ESI (Ethernet segment identifier);
- [0] – EthTag. В текущей версии ПО не используется;
- [48] – длина MAC-адреса;
- [e0:d9:e3:17:6b:40] – MAC-адрес, изученный на удаленном VTEP;
- [32] – длина IP-адреса;
- [192.168.2.2] – IP-адрес;
- 272 – полная длина маршрута.

Маршрут данного типа используется для передачи информации об изученных в пределах VXLAN соответствий MAC- и IP-адресов.

### Маршрут типа 2 (MAC), изученный за ethernet segment:

Route distinguisher: 10.0.0.2:2					
*>i [2] [03:11:22:33:44:55:66:00:03:e8] [0] [48] [e0:d9:e3:a8:45:40] [0] [0.0.0.0] /216					
	10.0.0.2	0	100	0	?

Где:

- [03:11:22:33:44:55:66:00:03:e8] – ESI (Ethernet segment identifier).

То же, что и обычный маршрут типа 2, только содержащий информацию об ethernet segment, за которым изучен MAC-адрес.

### Маршрут типа 3:

Network	Nexthop	Metric	LocPrf	Weight	Path
---------	---------	--------	--------	--------	------

```
Route distinguisher: 10.0.0.1:1000
*> [3][0][32][10.0.0.1]/88
      0.0.0.0      0      100      0      ?
```

Где:

- [3] – тип маршрута;
- [0] – EthTag. В текущей версии ПО не используется;
- [32] – длина IP-адреса источника маршрута;
- [10.0.0.1] – IP-адрес источника маршрута;
- [88] – полная длина маршрута.

Используется для приема и передачи BUM-трафика в пределах VXLAN.

#### Маршрут типа 4:

```
Route distinguisher: 10.0.0.1:0
*>i [4][03:11:22:33:44:55:66:00:03:e8][32][10.0.0.1]/136
      10.0.0.1      0      100      0      ?
```

Где:

- [4] – тип маршрута;
- [03:11:22:33:44:55:66:00:03:e8] – ESI (Ethernet segment identifier);
- [32] – длина IP-адреса источника маршрута;
- [10.0.0.1] – IP-адрес источника маршрута;
- 136 – полная длина маршрута.

Данный маршрут всегда создается один на сегмент, т.е. per ESI. Несет информацию о доступных с данного роутера Ethernet Segment-ах. Необходим для выбора DF (Designated Forwarder).

#### Маршрут типа 5:

```
Route distinguisher: 10.0.0.2:34564
*>i [5][0][0][24][192.168.2.0]/224
      10.0.0.2      0      100      0      ?
```

Где:

- [5] – тип маршрута;
- [0] – ESI (Ethernet segment identifier);
- [0] – EthTag. В текущей версии ПО не используется;
- [24] – длина префикса;
- [192.168.2.0] – IP-адрес сети назначения;
- 224 – полная длина маршрута.

Этот тип маршрута используется для распространения информации о сетях, доступных через тот или иной VTEP.

- **show evpn inclusive-multicast [word]**

Отображает информацию об имеющихся на устройстве маршрутах типа 3, которые используются для передачи BUM-трафика. Есть возможность фильтрации вывода по имени конкретной VXLAN.

Пример вывода:

```
Leaf_1#show evpn inclusive-multicast

VXLAN test_vxlan

IP Address      VNI      Source Tunnel Address  Remote Route
Distinguisher                                         Tunnel Type
```

```

-----
10.0.0.1      101000  Local  10.0.0.1      10.0.0.1:1000  Ingress Repl.
10.0.0.2      101000  Remote 10.0.0.2      10.0.0.2:1000  Ingress Repl.
10.0.0.3      101000  Remote 10.0.0.3      10.0.0.3:1000  Ingress Repl.

```

- **show evpn ingress-replication [vni vni-id]**

Выводит список VTEP, куда выполняется целевая рассылка BUM-трафика методом ingress-replication для всех созданных VNI. Есть возможность фильтрации вывода по номеру VNI.

Пример вывода:

```

Leaf_1#show evpn ingress-replication

* - Out of resources, flood to all VTEPs.

VNI      VLAN  List of VTEPs
-----
101000   1000  10.0.0.2
         10.0.0.3

101001   1001  10.0.0.2
         10.0.0.3

101002   1002  10.0.0.2
         10.0.0.3

```

Для VNI, которым не хватило системных ресурсов, в списке VTEPs будет выводиться символ \*. Пример:

```

Leaf_1#show evpn ingress-replication

* - Out of resources, flood to all VTEPs.

VNI      VLAN  List of VTEPs
-----
...
103456   3456  *
103457   3457  *
...

```

- **show evpn ingress-replication flood-domain**

Отображает список всех удаленных VTEP в IP-фабрике (т.е. кроме локального), а также список VNI/VLAN, попавших во flood-domain. При нехватке системных ресурсов для целевой рассылки BUM-трафика методом ingress-replication, VNI помещается во flood-domain. Для таких VNI рассылка BUM-трафика выполняется на все VTEP IP-фабрики.

Пример вывода:

```

Leaf_1#show evpn ingress-replication flood-domain

All connected VTEPs:

VTEPs
-----
10.0.0.2
10.0.0.3

Flooded to all connected VTEPs:

VNI      VLAN

```

```
-----
103456  3456
103457  3457
```

В примере в VNI 103456 и 103457 BUM-трафик будет рассылаться на все VTEP IP-фабрики вне зависимости от того, присутствуют ли на них указанные VNI.

- **show evpn ingress-replication resources**

Отображает информацию о системных ресурсах, использующихся для распространения BUM-трафика.

Пример вывода:

```
Leaf_1#show evpn ingress-replication resources

Replication Entries          In-Use    Free      Max
-----
Active Entries              8         6182     6190
  Flood domain Entries      2          127      129
  Active Entries            6         6055     6061

Inactive Entries            0         409400   409400
```

Где:

- Active Entries — активные записи, используемые в данный момент. Отражают степень использования системных ресурсов. Значения являются суммой двух следующих строк;
- Flood domain Entries — записи, необходимые для функционирования flood-домена (заполняемость зависит от количества VTEP);
- Active Entries (нижняя) — записи, используемые для целевой рассылки BUM-трафика методом ingress-replication;
- Inactive Entries — неактивные записи, которым не хватило места в системных ресурсах.

Когда количество свободных активных записей приблизится к нулю, новые VNI/VLAN начнут попадать во flood domain. Соответственно начнет расти счетчик неактивных записей. При освобождении активных записей будет запущен обратный процесс.

- **show evpn ethernet-segment {port-channel group | es\_number | mac-address | esi} [detailed]**

Отображает информацию об ethernet-сегментах. Есть возможность фильтрации вывода по номеру port-channel, номеру ES, системному MAC-адресу ES и идентификатору ESI.

Детальный вывод предоставляет подробную информацию.

Пример вывода:

```
Leaf_1#show evpn ethernet-segment

Interface Status      ES Number  Ethernet Segment ID (ESI)  Remote members
      admin/oper
-----
Po1      UP/UP       1000      03:11:22:33:44:55:66:00:03:e8  10.0.0.2
Po2      UP/DOWN    1001      03:11:22:33:44:55:77:00:03:e9
Po3      UP/DOWN    1002      03:11:22:33:44:55:88:00:03:ea
```

Пример вывода детальной информации:

```
Leaf_1#show evpn ethernet-segment 03:11:22:33:44:55:66:00:03:e8 detailed

Ethernet Segment: Po1
ESI: 03:11:22:33:44:55:66:00:03:e8
ES number is 1000
```

```

ES system MAC address is 11:22:33:44:55:66
Administrative status is up
Operational status is up

All-Active multi-homing mode
Route Distinguisher is 10.0.0.1:0 (auto-assigned)
Route Target is 1122.3344.5566 (auto-assigned)
Designated Forwarder election delay is 3 seconds
Members:

```

Router ID	Type
10.0.0.1	local
10.0.0.2	remote

Connected VXLANs:

VNI	VLAN ID	Designated Forwarder	Name
102	2	10.0.0.1	mcast2
103	3	10.0.0.2	mcast3
104	4	10.0.0.1	mcast4
105	5	10.0.0.2	mcast5

В выводе детальной информации можно увидеть настроенные ES number и ES system MAC address, а также сгенерированный на их основе ESI: 03:11:22:33:44:55:66:00:03:e8.

Кроме того, можно видеть информацию обо всех устройствах-членах Ethernet-сегмента. В данном случае о Router ID 10.0.0.1 и 10.0.0.2.

Выбор Designated Forwarder может отличаться для различных VXLAN, что можно видеть в таблице Connected VXLANs. Именно DF отвечает за отправку BUM-трафика в конкретный Ethernet-сегмент в пределах конкретной VLAN.

- **show evpn ethernet-segment multihoming core-tracking**

Отображает статус функционала evpn multihoming core-tracking, delay timer, состояние участвующих BGP-соседей и имеющиеся ethernet-сегменты.

```
Leaf_1#show evpn ethernet-segment multihoming core-tracking
```

```
EVPN Core status is CONNECTED
```

```
Multihoming ES UP delay timer is disabled
```

```
Peer trackings:
```

BGP neighbor	State
10.0.1.1	ESTABLISHED
10.0.2.2	IDLE

```
Ethernet segments:
```

Interface	Status	Ethernet Segment ID (ESI)
	admin/oper	

В данном примере core-tracking включен для двух BGP-соседей, один из которых в данный момент не активен.

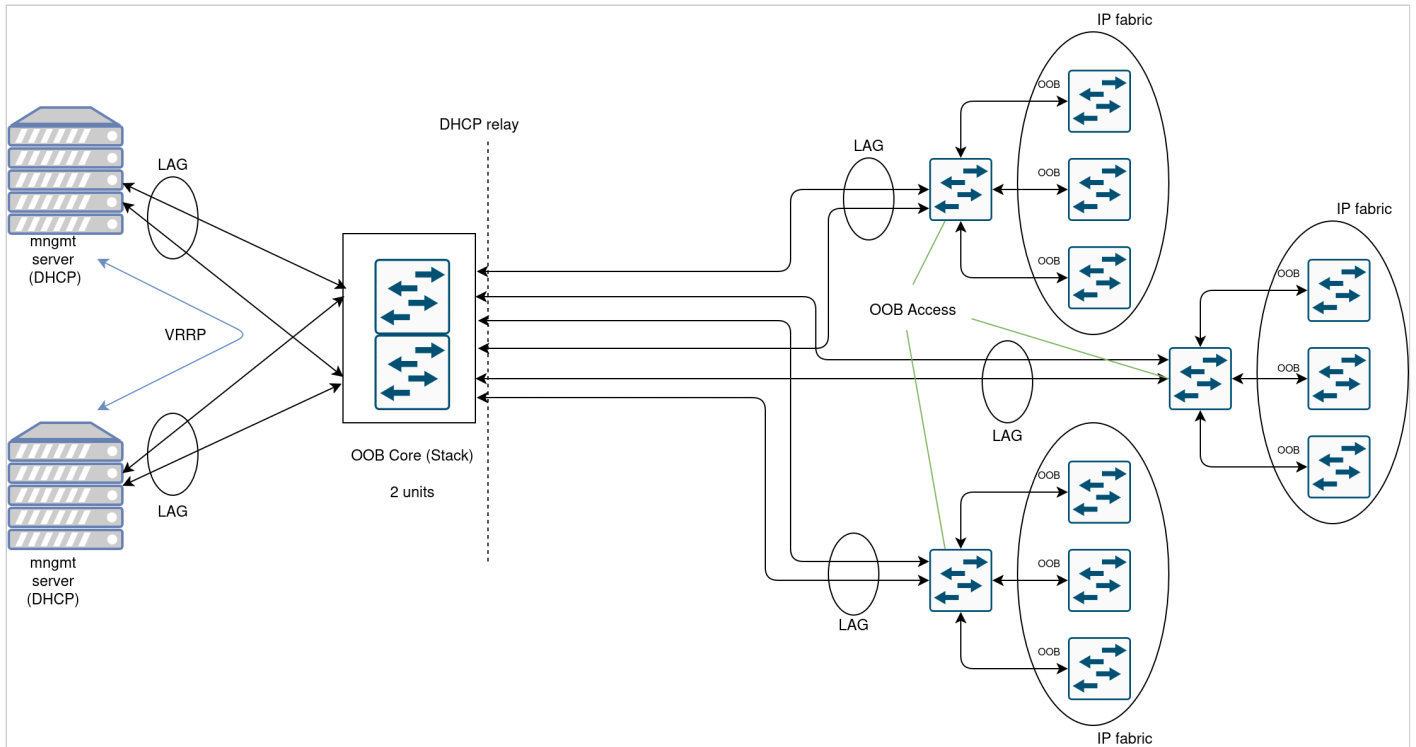
Delay timer не задан.

Показан подключенный ethernet-сегмент и его статус.

## 15 Сеть управления

Подход out-of-band management (внеполосное управление) подразумевает передачу управляющей информации отдельно от передачи данных. Применимо к IP-фабрике это означает организацию сети, обеспечивающей доступ к OOB-интерфейсам всех её (фабрики) устройств и при этом не пересекающейся с самой сетью IP-фабрики. Такой подход позволяет обеспечить управление устройствами вне зависимости от состояния сети, передающей коммерческую информацию.

### 15.1 Схема сети OOB



- OOB core – стек из коммутаторов агрегации, выполняющий функции маршрутизации и DHCP relay. Служит для подключения к сети управления коммутаторов OOB access посредством LAG для повышения отказоустойчивости.
- Коммутаторы OOB access выполняют роль коммутаторов доступа к сети управления, обеспечивая подключение к ней управляемых устройств (IP-фабрики) посредством OOB-интерфейсов.

Используемые в схеме протоколы и технологии:

- LAG (link aggregation group) – группы агрегации каналов. Каждая группа портов должна состоять из интерфейсов Ethernet с одинаковой скоростью, работающих в дуплексном режиме. Объединение портов в группу увеличивает пропускную способность канала между взаимодействующими устройствами и повышает отказоустойчивость. Группа портов является для коммутатора одним логическим портом.
- LACP (link aggregation control protocol) – позволяет объединять несколько физических каналов в один, таким образом создавая вышеописанную LAG.
- Протокол VRRP (Virtual Router Redundancy Protocol) – позволяет зарезервировать шлюз по умолчанию, который используют все устройства IP-фабрики для обмена информацией с системой управления, мониторинга, синхронизации времени и т.д.
- DHCP Relay агент. Задачей DHCP Relay агента является передача DHCP-пакетов от клиента к серверу и обратно в случае, если DHCP-сервер находится в одной сети, а клиент – в другой. Другой функцией является добавление дополнительных опций в DHCP-запросы клиента (например, опции 82).

- Stack – коммутаторы OOB core объединены в стек. Стекирование позволяет им функционировать как единое устройство, тем самым повышая отказоустойчивость и облегчая управление. Рекомендуется использовать кольцевую топологию для повышения отказоустойчивости стека.

## 15.2 Конфигурации устройств

### 15.2.1 OOB core

```
no spanning-tree
!
vlan database
vlan 2,111
exit
!
ip dhcp relay address 192.168.11.1
ip dhcp relay enable
ip dhcp snooping
ip dhcp snooping vlan 2
!
hostname OOB_Core
!
interface TenGigabitEthernet1/0/1
channel-group 1 mode auto
exit
!
interface TenGigabitEthernet1/0/2
channel-group 2 mode auto
exit
!
interface TenGigabitEthernet1/0/3
channel-group 3 mode auto
exit
!
interface TenGigabitEthernet1/0/11
description Server
switchport access vlan 111
exit
!
interface TenGigabitEthernet2/0/1
channel-group 1 mode auto
exit
!
interface TenGigabitEthernet2/0/2
channel-group 2 mode auto
exit
!
interface TenGigabitEthernet2/0/3
channel-group 3 mode auto
exit
!
interface TenGigabitEthernet2/0/11
description Server
switchport access vlan 111
exit
!
interface Port-Channel1
description OOB_access_1
switchport mode trunk
switchport trunk allowed vlan add 2
switchport forbidden default-vlan
exit
```

```
!  
interface Port-Channel2  
description OOB_access_2  
switchport mode trunk  
switchport trunk allowed vlan add 2  
switchport forbidden default-vlan  
exit  
!  
interface Port-Channel3  
description OOB_access_3  
switchport mode trunk  
switchport trunk allowed vlan add 2  
switchport forbidden default-vlan  
exit  
!  
interface vlan 1  
shutdown  
exit  
!  
interface vlan 2  
ip address 192.168.50.1 255.255.255.0  
ip dhcp relay enable  
exit  
!  
interface vlan 111  
ip address 192.168.11.10 255.255.255.0  
exit  
!  
!  
end
```

## 15.2.2 OOB access

```
no spanning-tree
!
vlan database
vlan 2
exit
!
hostname OOB_Access_1
!
interface gigabitethernet1/0/1
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/2
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/3
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/4
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/5
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/6
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/7
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/8
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/9
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/10
description OOB
switchport access vlan 2
```

```
exit
!
interface gigabitethernet1/0/11
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/12
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/13
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/14
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/15
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/16
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/17
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/18
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/19
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/20
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/21
description 00B
switchport access vlan 2
exit
!
interface gigabitethernet1/0/22
description 00B
switchport access vlan 2
```

```
exit
!
interface gigabitethernet1/0/23
description OOB
switchport access vlan 2
exit
!
interface gigabitethernet1/0/24
description OOB
switchport access vlan 2
exit
!
interface tengigabitethernet1/0/1
channel-group 1 mode auto
exit
!
interface tengigabitethernet1/0/2
channel-group 1 mode auto
exit
!
interface Port-channel1
description OOB_Core
switchport mode trunk
switchport trunk allowed vlan add 2
switchport forbidden default-vlan
exit
!
interface vlan 1
shutdown
exit
!
!
end
```

## 16 Приложение 1

В приложении содержатся полные конфигурации устройств, используемые в данном руководстве.

### 16.1 Конфигурации с использованием протокола OSPF

#### 16.1.1 Spine\_1

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_1
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.2 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.2 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.2 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.1.1 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.1.2 area 0.0.0.0
  network 172.16.2.2 area 0.0.0.0
  network 172.16.3.2 area 0.0.0.0
  router-id 10.0.1.1
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.3.2
  ip ospf network point-to-point
exit
!
router bgp 65500
```

```
bgp router-id 10.0.1.1
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group LEAF_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
  route-reflector-client
exit
!
neighbor 10.0.0.1
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.0.2
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.0.3
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
!
end
```

## 16.1.2 Spine\_2

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_2
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.6 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.6 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.6 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.2.2 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.1.6 area 0.0.0.0
  network 172.16.2.6 area 0.0.0.0
  network 172.16.3.6 area 0.0.0.0
  router-id 10.0.2.2
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.6
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.6
  ip ospf network point-to-point
exit
!
interface ip 172.16.3.6
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.2.2
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
```

```
!  
peer-group LEAF_GROUP  
  remote-as 65500  
  update-source loopback 1  
  fall-over bfd  
  route-reflector-client  
exit  
!  
neighbor 10.0.0.1  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.2  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.3  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
exit  
!  
!  
end
```

### 16.1.3 Leaf\_1

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_1
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.1.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.1.5 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.0.1 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.1.1 area 0.0.0.0
  network 172.16.1.5 area 0.0.0.0
  router-id 10.0.0.1
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.1
  ip ospf network point-to-point
exit
!
interface ip 172.16.1.5
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.0.1
```

```
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
exit
!
!
end
```

## 16.1.4 Leaf\_2

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_2
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.2.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.2.5 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.0.2 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.2.1 area 0.0.0.0
  network 172.16.2.5 area 0.0.0.0
  router-id 10.0.0.2
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.2.1
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.5
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.0.2
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
```

```
!  
peer-group SPINE_GROUP  
  remote-as 65500  
  update-source loopback 1  
  fall-over bfd  
exit  
!  
neighbor 10.0.1.1  
  peer-group SPINE_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.2.2  
  peer-group SPINE_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
exit  
!  
!  
end
```

## 16.1.5 Leaf\_3

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_3
!
interface TenGigabitEthernet1/0/11
  description Host2
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.3.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.3.5 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.0.3 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.3.1 area 0.0.0.0
  network 172.16.3.5 area 0.0.0.0
  router-id 10.0.0.3
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.3.1
  ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.0.3
```

```
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
exit
!
!
end
```

## 16.2 Конфигурации с использованием протокола IS-IS

### 16.2.1 Spine\_1

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_1
!
interface HundredGigabitEthernet1/0/1
description Leaf_1
ip address 172.16.1.2 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/2
description Leaf_2
ip address 172.16.2.2 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/3
description Leaf_3
ip address 172.16.3.2 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface loopback1
ip address 10.0.1.1 255.255.255.255
exit
!
!
router bgp 65500
  bgp router-id 10.0.1.1
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group LEAF_GROUP
    remote-as 65500
    update-source loopback 1
    fall-over bfd
    route-reflector-client
  exit
  !
  neighbor 10.0.0.1
    peer-group LEAF_GROUP
    address-family ipv4 unicast
  exit
```

```
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.0.2
peer-group LEAF_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
neighbor 10.0.0.3
peer-group LEAF_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
router isis
address-family ipv4 unicast
redistribute connected
exit
net 49.0001.1111.1111.1111.00
exit
!
!
end
```

## 16.2.2 Spine\_2

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_2
!
interface HundredGigabitEthernet1/0/1
description Leaf_1
ip address 172.16.1.6 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/2
description Leaf_2
ip address 172.16.2.6 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/3
description Leaf_3
ip address 172.16.3.6 255.255.255.252
ip router isis
isis network point-to-point
exit
!
interface loopback1
ip address 10.0.2.2 255.255.255.255
exit
!
!
router bgp 65500
  bgp router-id 10.0.2.2
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group LEAF_GROUP
    remote-as 65500
    update-source loopback 1
    fall-over bfd
    route-reflector-client
  exit
  !
  neighbor 10.0.0.1
    peer-group LEAF_GROUP
    address-family ipv4 unicast
    exit
    !
    address-family l2vpn evpn
    exit
```

```
exit
!
neighbor 10.0.0.2
peer-group LEAF_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.0.3
peer-group LEAF_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
router isis
address-family ipv4 unicast
redistribute connected
exit
net 49.0001.2222.2222.2222.00
exit
!
!
end
```

### 16.2.3 Leaf\_1

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_1
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.1.1 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.1.5 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface loopback1
  ip address 10.0.0.1 255.255.255.255
exit
!
!
router bgp 65500
  bgp router-id 10.0.0.1
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
```

```
neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
exit
!
router isis
  address-family ipv4 unicast
  redistribute connected
  exit
  net 49.0001.0001.0001.0001.00
exit
!
!
end
```

## 16.2.4 Leaf\_2

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_2
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.2.1 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.2.5 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface loopback1
  ip address 10.0.0.2 255.255.255.255
exit
!
!
router bgp 65500
  bgp router-id 10.0.0.2
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
  exit
  !
  neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
```

```
    address-family l2vpn evpn
    exit
  exit
  !
  neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  exit
  exit
  !
  router isis
  address-family ipv4 unicast
  redistribute connected
  exit
  net 49.0001.0002.0002.0002.00
  exit
  !
  !
  end
```

## 16.2.5 Leaf\_3

```
no spanning-tree
!
vlan database
  vlan 1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_3
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.3.1 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.3.5 255.255.255.252
  ip router isis
  isis network point-to-point
exit
!
interface loopback1
  ip address 10.0.0.3 255.255.255.255
exit
!
!
router bgp 65500
  bgp router-id 10.0.0.3
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
```

```
neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
!
neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
exit
exit
!
router isis
  address-family ipv4 unicast
  redistribute connected
  exit
  net 49.0001.0003.0003.0003.00
exit
!
!
end
```

## 16.3 Конфигурации для multicast VXLAN

### 16.3.1 Spine\_1

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_1
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.2 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.2 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.2 255.255.255.252
  ip pim
exit
!
interface loopback1
  ip address 10.0.1.1 255.255.255.255
exit
!
interface loopback2
  ip address 10.100.100.100 255.255.255.255
  description Anycast_RP_IP
exit
!
!
router ospf 1
  network 172.16.1.2 area 0.0.0.0
  network 172.16.2.2 area 0.0.0.0
  network 172.16.3.2 area 0.0.0.0
  router-id 10.0.1.1
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.2
  ip ospf network point-to-point
exit
!
```

```

interface ip 172.16.3.2
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.1.1
 address-family ipv4 unicast
 exit
 !
 address-family l2vpn evpn
 exit
 !
 peer-group LEAF_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
  route-reflector-client
 exit
 !
 neighbor 10.0.0.1
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.2
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.3
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
router msdp
 connect-source 10.0.1.1
 originator-ip 10.100.100.100
 !
 peer 10.0.2.2
  mesh-group TESTGR
 exit
exit

```

```
!  
end
```

### 16.3.2 Spine\_2

```
no spanning-tree  
!  
port jumbo-frame  
!  
ip maximum-paths 32  
!  
hostname Spine_2  
!  
interface HundredGigabitEthernet1/0/1  
  description Leaf_1  
  ip address 172.16.1.6 255.255.255.252  
  ip pim  
exit  
!  
interface HundredGigabitEthernet1/0/2  
  description Leaf_2  
  ip address 172.16.2.6 255.255.255.252  
  ip pim  
exit  
!  
interface HundredGigabitEthernet1/0/3  
  description Leaf_3  
  ip address 172.16.3.6 255.255.255.252  
  ip pim  
exit  
!  
interface loopback1  
  ip address 10.0.2.2 255.255.255.255  
exit  
!  
interface loopback2  
  ip address 10.100.100.100 255.255.255.255  
  description Anycast_RP_IP  
exit  
!  
!  
router ospf 1  
  network 172.16.1.6 area 0.0.0.0  
  network 172.16.2.6 area 0.0.0.0  
  network 172.16.3.6 area 0.0.0.0  
  router-id 10.0.2.2  
  timers spf delay 0  
  redistribute connected subnets  
exit  
!  
interface ip 172.16.1.6  
  ip ospf network point-to-point  
exit  
!  
interface ip 172.16.2.6  
  ip ospf network point-to-point  
exit  
!  
interface ip 172.16.3.6
```

```

 ip ospf network point-to-point
 exit
 !
 router bgp 65500
  bgp router-id 10.0.2.2
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  !
  peer-group LEAF_GROUP
   remote-as 65500
   update-source loopback 1
   fall-over bfd
   route-reflector-client
  exit
  !
  neighbor 10.0.0.1
   peer-group LEAF_GROUP
   address-family ipv4 unicast
   exit
   !
   address-family l2vpn evpn
   exit
  exit
  !
  neighbor 10.0.0.2
   peer-group LEAF_GROUP
   address-family ipv4 unicast
   exit
   !
   address-family l2vpn evpn
   exit
  exit
  !
  neighbor 10.0.0.3
   peer-group LEAF_GROUP
   address-family ipv4 unicast
   exit
   !
   address-family l2vpn evpn
   exit
  exit
 exit
 !
 !
 ip multicast-routing pim
 !
 ip pim rp-address 10.100.100.100
 !
 router msdp
  connect-source 10.0.2.2
  originator-ip 10.100.100.100
  !
  peer 10.0.1.1
   mesh-group TESTGR
  exit
 exit
 !

```

```
end
```

### 16.3.3 Leaf\_1

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_1
!
interface TenGigabitEthernet1/0/9
  description Host1_mcast
  switchport mode trunk
  switchport trunk allowed vlan add 2-5
  switchport forbidden default-vlan
exit
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
```

```

description Spine_1
ip address 172.16.1.1 255.255.255.252
ip pim
exit
!
interface HundredGigabitEthernet1/0/2
description Spine_2
ip address 172.16.1.5 255.255.255.252
ip pim
exit
!
interface loopback1
ip address 10.0.0.1 255.255.255.255
ip pim
ip igmp static-group 233.0.0.2
ip igmp static-group 233.0.0.3
ip igmp static-group 233.0.0.4
ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
network 172.16.1.1 area 0.0.0.0
network 172.16.1.5 area 0.0.0.0
router-id 10.0.0.1
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.1.1
ip ospf network point-to-point
exit
!
interface ip 172.16.1.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.1
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!

```

```
neighbor 10.0.2.2
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
ip multicast multipath group-paths-num
!
end
```

### 16.3.4 Leaf\_2

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_2
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.2.1 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.2.5 255.255.255.252
  ip pim
exit
!
interface loopback1
  ip address 10.0.0.2 255.255.255.255
```

```

ip pim
ip igmp static-group 233.0.0.2
ip igmp static-group 233.0.0.3
ip igmp static-group 233.0.0.4
ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
network 172.16.2.1 area 0.0.0.0
network 172.16.2.5 area 0.0.0.0
router-id 10.0.0.2
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.2.1
ip ospf network point-to-point
exit
!
interface ip 172.16.2.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.2
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.2.2
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
!
ip multicast-routing pim
!

```

```
ip pim rp-address 10.100.100.100
!  
ip multicast multipath group-paths-num  
!  
end
```

### 16.3.5 Leaf\_3

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_3
!
interface TenGigabitEthernet1/0/9
  description Host2_mcast
  switchport mode trunk
  switchport trunk allowed vlan add 2-5
  switchport forbidden default-vlan
exit
!
interface TenGigabitEthernet1/0/11
  description Host2
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
```

```

ip address 172.16.3.1 255.255.255.252
ip pim
exit
!
interface HundredGigabitEthernet1/0/2
description Spine_2
ip address 172.16.3.5 255.255.255.252
ip pim
exit
!
interface loopback1
ip address 10.0.0.3 255.255.255.255
ip pim
ip igmp static-group 233.0.0.2
ip igmp static-group 233.0.0.3
ip igmp static-group 233.0.0.4
ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
network 172.16.3.1 area 0.0.0.0
network 172.16.3.5 area 0.0.0.0
router-id 10.0.0.3
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.3.1
ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.3
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.2.2

```

```
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
ip multicast multipath group-paths-num
!
end
```

## 16.4 Конфигурации для Symmetric IRB

### 16.4.1 Spine\_1

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_1
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.2 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.2 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.2 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.1.1 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.1.2 area 0.0.0.0
  network 172.16.2.2 area 0.0.0.0
  network 172.16.3.2 area 0.0.0.0
  router-id 10.0.1.1
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.3.2
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.1.1
  address-family ipv4 unicast
  exit
```

```
!  
address-family l2vpn evpn  
exit  
!  
peer-group LEAF_GROUP  
  remote-as 65500  
  update-source loopback 1  
  fall-over bfd  
  route-reflector-client  
exit  
!  
neighbor 10.0.0.1  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.2  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.3  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
exit  
!  
!  
end
```

## 16.4.2 Spine\_2

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_2
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.6 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.6 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.6 255.255.255.252
exit
!
interface loopback1
  ip address 10.0.2.2 255.255.255.255
exit
!
!
router ospf 1
  network 172.16.1.6 area 0.0.0.0
  network 172.16.2.6 area 0.0.0.0
  network 172.16.3.6 area 0.0.0.0
  router-id 10.0.2.2
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.6
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.6
  ip ospf network point-to-point
exit
!
interface ip 172.16.3.6
  ip ospf network point-to-point
exit
!
router bgp 65500
  bgp router-id 10.0.2.2
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
```

```
!  
peer-group LEAF_GROUP  
  remote-as 65500  
  update-source loopback 1  
  fall-over bfd  
  route-reflector-client  
exit  
!  
neighbor 10.0.0.1  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.2  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
!  
neighbor 10.0.0.3  
  peer-group LEAF_GROUP  
  address-family ipv4 unicast  
  exit  
  !  
  address-family l2vpn evpn  
  exit  
exit  
exit  
!  
!  
end
```

### 16.4.3 Leaf\_1

```
ip vrf VRF1
  vni 100100
  route-target both 65500:100100
exit
!
!
no spanning-tree
!
vlan database
  vlan 100,1000
exit
!
port jumbo-frame
!
vxlan L3_vxlan
  vni 100100 ip-routing
  vlan 100
exit
!
vxlan test_vxlan1
  vni 101000
  vlan 1000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_1
!
line console
  exec-timeout 0
exit
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.1.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.1.5 255.255.255.252
exit
!
interface vlan 100
  ip vrf VRF1
exit
!
interface vlan 1000
  ip vrf VRF1
  ip address 192.168.1.1 255.255.255.0
exit
```

```

!
interface loopback1
 ip address 10.0.0.1 255.255.255.255
exit
!
!
!
!
router ospf 1
 network 172.16.1.1 area 0.0.0.0
 network 172.16.1.5 area 0.0.0.0
 router-id 10.0.0.1
 timers spf delay 0
 redistribute connected subnets
exit
!
interface ip 172.16.1.1
 ip ospf network point-to-point
exit
!
interface ip 172.16.1.5
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.0.1
 address-family ipv4 unicast
 exit
!
 address-family l2vpn evpn
 exit
!
 peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
 neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
exit
!
 neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
exit
!
vrf VRF1
 address-family ipv4 unicast
  redistribute connected
 exit

```

```
exit  
exit  
!  
!  
end
```

## 16.4.4 Leaf\_2

```
ip vrf VRF1
  vni 100100
  route-target both 65500:100100
exit
!
!
no spanning-tree
!
vlan database
  vlan 100,2000
exit
!
port jumbo-frame
!
vxlan L3_vxlan
  vni 100100 ip-routing
  vlan 100
exit
!
vxlan test_vxlan2
  vni 102000
  vlan 2000
exit
!
!
ip maximum-paths 32
!
anycast-gateway mac-address 00:00:00:11:11:11
!
hostname Leaf_2
!
line console
  exec-timeout 0
exit
!
interface TenGigabitEthernet1/0/11
  description Host2
  switchport access vlan 2000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.2.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.2.5 255.255.255.252
exit
!
interface vlan 100
  ip vrf VRF1
exit
!
interface vlan 2000
  ip vrf VRF1
```

```

ip address 192.168.2.1 255.255.255.0
anycast-gateway
exit
!
interface loopback1
ip address 10.0.0.2 255.255.255.255
exit
!
!
!
!
router ospf 1
network 172.16.2.1 area 0.0.0.0
network 172.16.2.5 area 0.0.0.0
router-id 10.0.0.2
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.2.1
ip ospf network point-to-point
exit
!
interface ip 172.16.2.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.2
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.2.2
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
vrf VRF1

```

```
address-family ipv4 unicast
  redistribute connected
exit
exit
exit
!
!
end
```

## 16.4.5 Leaf\_3

```
ip vrf VRF1
  vni 100100
  route-target both 65500:100100
exit
!
!
no spanning-tree
!
vlan database
  vlan 100,2000
exit
!
port jumbo-frame
!
vxlan L3_vxlan
  vni 100100 ip-routing
  vlan 100
exit
!
vxlan test_vxlan2
  vni 102000
  vlan 2000
exit
!
!
ip maximum-paths 32
!
anycast-gateway mac-address 00:00:00:11:11:11
!
hostname Leaf_3
!
interface TenGigabitEthernet1/0/11
  description Host3
  switchport access vlan 2000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.3.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.3.5 255.255.255.252
exit
!
interface vlan 100
  ip vrf VRF1
exit
!
interface vlan 2000
  ip vrf VRF1
  ip address 192.168.2.1 255.255.255.0
  anycast-gateway
exit
!
```

```

interface loopback1
 ip address 10.0.0.3 255.255.255.255
exit
!
!
!
!
router ospf 1
 network 172.16.3.1 area 0.0.0.0
 network 172.16.3.5 area 0.0.0.0
 router-id 10.0.0.3
 timers spf delay 0
 redistribute connected subnets
exit
!
interface ip 172.16.3.1
 ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.0.3
 address-family ipv4 unicast
 exit
 !
 address-family l2vpn evpn
 exit
 !
 peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
 exit
 !
 neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 vrf VRF1
  address-family ipv4 unicast
  redistribute connected
  exit
 exit

```

```
exit
!  
!  
end
```

### 16.4.6 Leaf\_3 + OSPF

```
ip vrf VRF1
  vni 100100
  route-target both 65500:100100
exit
!  
!  
no spanning-tree
!  
vlan database
  vlan 100,2000
exit
!  
port jumbo-frame
!  
vxlan L3_vxlan
  vni 100100 ip-routing
  vlan 100
exit
!  
vxlan test_vxlan2
  vni 102000
  vlan 2000
exit
!  
!  
ip maximum-paths 32
!  
hostname Leaf_3
!  
line console
  exec-timeout 0
exit
!  
interface TenGigabitEthernet1/0/11
  description Host3
  switchport access vlan 2000
exit
!  
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.3.1 255.255.255.252
exit
!  
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.3.5 255.255.255.252
exit
!  
interface vlan 100
  ip vrf VRF1
exit
!  
!
```

```

interface vlan 2000
 ip vrf VRF1
 ip address 192.168.2.1 255.255.255.0
exit
!
interface loopback1
 ip address 10.0.0.3 255.255.255.255
exit
!
!
!
!
router ospf 1
 network 172.16.3.1 area 0.0.0.0
 network 172.16.3.5 area 0.0.0.0
 router-id 10.0.0.3
 timers spf delay 0
 redistribute connected subnets
exit
!
router ospf 2 vrf VRF1
 network 192.168.2.1 area 10.10.10.10
 router-id 192.168.2.1
 redistribute bgp subnets
exit
!
interface ip 172.16.3.1
 ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.0.3
 address-family ipv4 unicast
 exit
 !
 address-family l2vpn evpn
 exit
 !
 peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
 exit
 !
 neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast

```

```
exit
!  
address-family l2vpn evpn  
exit  
exit  
!  
vrf VRF1  
address-family ipv4 unicast  
redistribute connected  
redistribute ospf 2 match internal  
redistribute ospf 2 match external-1  
redistribute ospf 2 match external-2  
exit  
exit  
exit  
!  
!  
end
```

## 16.4.7 Leaf\_3 + eBGP

```
ip vrf VRF1
  vni 100100
  route-target both 65500:100100
exit
!
!
no spanning-tree
!
vlan database
  vlan 100,2000
exit
!
port jumbo-frame
!
vxlan L3_vxlan
  vni 100100 ip-routing
  vlan 100
exit
!
vxlan test_vxlan2
  vni 102000
  vlan 2000
exit
!
!
ip maximum-paths 32
!
hostname Leaf_3
!
line console
  exec-timeout 0
exit
!
interface TenGigabitEthernet1/0/11
  description Host3
  switchport access vlan 2000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.3.1 255.255.255.252
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.3.5 255.255.255.252
exit
!
interface vlan 100
  ip vrf VRF1
exit
!
interface vlan 2000
  ip vrf VRF1
  ip address 192.168.2.1 255.255.255.0
exit
```

```

!
interface loopback1
 ip address 10.0.0.3 255.255.255.255
exit
!
!
!
!
router ospf 1
 network 172.16.3.1 area 0.0.0.0
 network 172.16.3.5 area 0.0.0.0
 router-id 10.0.0.3
 timers spf delay 0
 redistribute connected subnets
exit
!
interface ip 172.16.3.1
 ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.0.3
 address-family ipv4 unicast
 exit
!
 address-family l2vpn evpn
 exit
!
 peer-group SPINE_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
exit
!
 neighbor 10.0.1.1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
exit
!
 neighbor 10.0.2.2
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
!
  address-family l2vpn evpn
  exit
exit
!
vrf VRF1
 address-family ipv4 unicast
  redistribute connected
  redistribute bgp

```

```
exit
!  
neighbor 192.168.2.2  
  remote-as 65600  
  address-family ipv4 unicast  
exit  
exit  
exit  
!  
!  
end
```

## 16.5 Конфигурации для EVPN multihoming

### 16.5.1 Spine\_1

```
no spanning-tree
!
port jumbo-frame
!
ip maximum-paths 32
!
hostname Spine_1
!
interface HundredGigabitEthernet1/0/1
  description Leaf_1
  ip address 172.16.1.2 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/2
  description Leaf_2
  ip address 172.16.2.2 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/3
  description Leaf_3
  ip address 172.16.3.2 255.255.255.252
  ip pim
exit
!
interface loopback1
  ip address 10.0.1.1 255.255.255.255
exit
!
interface loopback2
  ip address 10.100.100.100 255.255.255.255
  description Anycast_RP_IP
exit
!
!
router ospf 1
  network 172.16.1.2 area 0.0.0.0
  network 172.16.2.2 area 0.0.0.0
  network 172.16.3.2 area 0.0.0.0
  router-id 10.0.1.1
  timers spf delay 0
  redistribute connected subnets
exit
!
interface ip 172.16.1.2
  ip ospf network point-to-point
exit
!
interface ip 172.16.2.2
  ip ospf network point-to-point
exit
!
```

```

interface ip 172.16.3.2
 ip ospf network point-to-point
exit
!
router bgp 65500
 bgp router-id 10.0.1.1
 address-family ipv4 unicast
 exit
 !
 address-family l2vpn evpn
 exit
 !
 peer-group LEAF_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
  route-reflector-client
 exit
 !
 neighbor 10.0.0.1
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.2
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.3
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
router msdp
 connect-source 10.0.1.1
 originator-ip 10.100.100.100
 !
 peer 10.0.2.2
  mesh-group TESTGR
 exit
exit

```

```
!  
end
```

## 16.5.2 Spine\_2

```
no spanning-tree  
!  
port jumbo-frame  
!  
ip maximum-paths 32  
!  
hostname Spine_2  
!  
interface HundredGigabitEthernet1/0/1  
  description Leaf_1  
  ip address 172.16.1.6 255.255.255.252  
  ip pim  
exit  
!  
interface HundredGigabitEthernet1/0/2  
  description Leaf_2  
  ip address 172.16.2.6 255.255.255.252  
  ip pim  
exit  
!  
interface HundredGigabitEthernet1/0/3  
  description Leaf_3  
  ip address 172.16.3.6 255.255.255.252  
  ip pim  
exit  
!  
interface loopback1  
  ip address 10.0.2.2 255.255.255.255  
exit  
!  
interface loopback2  
  ip address 10.100.100.100 255.255.255.255  
  description Anycast_RP_IP  
exit  
!  
!  
router ospf 1  
  network 172.16.1.6 area 0.0.0.0  
  network 172.16.2.6 area 0.0.0.0  
  network 172.16.3.6 area 0.0.0.0  
  router-id 10.0.2.2  
  timers spf delay 0  
  redistribute connected subnets  
exit  
!  
interface ip 172.16.1.6  
  ip ospf network point-to-point  
exit  
!  
interface ip 172.16.2.6  
  ip ospf network point-to-point  
exit  
!  
interface ip 172.16.3.6
```

```

 ip ospf network point-to-point
 exit
 !
router bgp 65500
 bgp router-id 10.0.2.2
 address-family ipv4 unicast
 exit
 !
 address-family l2vpn evpn
 exit
 !
 peer-group LEAF_GROUP
  remote-as 65500
  update-source loopback 1
  fall-over bfd
  route-reflector-client
 exit
 !
 neighbor 10.0.0.1
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.2
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 !
 neighbor 10.0.0.3
  peer-group LEAF_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
 exit
 exit
 !
 !
 ip multicast-routing pim
 !
 ip pim rp-address 10.100.100.100
 !
router msdp
 connect-source 10.0.2.2
 originator-ip 10.100.100.100
 !
 peer 10.0.1.1
  mesh-group TESTGR
 exit
 exit
 !

```

```
end
```

### 16.5.3 Leaf\_1

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_1
!
interface TenGigabitEthernet1/0/9
  description Host1_mcast
  switchport mode trunk
  switchport trunk allowed vlan add 2-5
  switchport forbidden default-vlan
exit
!
interface TenGigabitEthernet1/0/11
  description Host1
  switchport access vlan 1000
exit
!
interface TenGigabitEthernet1/0/12
```

```

    channel-group 1 mode auto
exit
!
interface HundredGigabitEthernet1/0/1
    description Spine_1
    ip address 172.16.1.1 255.255.255.252
    ip pim
exit
!
interface HundredGigabitEthernet1/0/2
    description Spine_2
    ip address 172.16.1.5 255.255.255.252
    ip pim
exit
!
interface Port-Channel1
    description Host1_LAG
    switchport mode trunk
    switchport trunk allowed vlan add 2-5
    hold-time up 60
    ethernet-segment 1000
        system-mac 11:22:33:44:55:66
    exit
exit
!
interface loopback1
    ip address 10.0.0.1 255.255.255.255
    ip pim
    ip igmp static-group 233.0.0.2
    ip igmp static-group 233.0.0.3
    ip igmp static-group 233.0.0.4
    ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
    network 172.16.1.1 area 0.0.0.0
    network 172.16.1.5 area 0.0.0.0
    router-id 10.0.0.1
    timers spf delay 0
    redistribute connected subnets
exit
!
interface ip 172.16.1.1
    ip ospf network point-to-point
exit
!
interface ip 172.16.1.5
    ip ospf network point-to-point
exit
!
router bgp 65500
    bgp router-id 10.0.0.1
    address-family ipv4 unicast
    exit
    !
    address-family l2vpn evpn
    exit
    !
    peer-group SPINE_GROUP

```

```
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
advertisement-interval 1 withdraw 1
peer-group SPINE_GROUP
evpn multihoming core-tracking
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!
neighbor 10.0.2.2
advertisement-interval 1 withdraw 1
peer-group SPINE_GROUP
evpn multihoming core-tracking
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
ip multicast multipath group-paths-num
!
end
```

## 16.5.4 Leaf\_2

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_2
!
interface TenGigabitEthernet1/0/12
  channel-group 1 mode auto
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
  ip address 172.16.2.1 255.255.255.252
  ip pim
exit
!
interface HundredGigabitEthernet1/0/2
  description Spine_2
  ip address 172.16.2.5 255.255.255.252
  ip pim
```

```

exit
!
interface Port-Channel1
description Host1_LAG
switchport mode trunk
switchport trunk allowed vlan add 2-5
hold-time up 60
ethernet-segment 1000
system-mac 11:22:33:44:55:66
exit
exit
!
interface loopback1
ip address 10.0.0.2 255.255.255.255
ip pim
ip igmp static-group 233.0.0.2
ip igmp static-group 233.0.0.3
ip igmp static-group 233.0.0.4
ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
network 172.16.2.1 area 0.0.0.0
network 172.16.2.5 area 0.0.0.0
router-id 10.0.0.2
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.2.1
ip ospf network point-to-point
exit
!
interface ip 172.16.2.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.2
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
advertisement-interval 1 withdraw 1
peer-group SPINE_GROUP
evpn multihoming core-tracking
address-family ipv4 unicast
exit
!
address-family l2vpn evpn

```

```
    exit
  exit
  !
  neighbor 10.0.2.2
  advertisement-interval 1 withdraw 1
  peer-group SPINE_GROUP
  evpn multihoming core-tracking
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  exit
  exit
  !
  !
  ip multicast-routing pim
  !
  ip pim rp-address 10.100.100.100
  !
  ip multicast multipath group-paths-num
  !
  end
```

## 16.5.5 Leaf\_3

```
no spanning-tree
!
vlan database
  vlan 2-5,1000
exit
!
port jumbo-frame
!
vxlan test_vxlan
  vni 101000
  vlan 1000
exit
!
vxlan mcast2
  vni 102
  vlan 2
  mcast-group 233.0.0.2
exit
!
vxlan mcast3
  vni 103
  vlan 3
  mcast-group 233.0.0.3
exit
!
vxlan mcast4
  vni 104
  vlan 4
  mcast-group 233.0.0.4
exit
!
vxlan mcast5
  vni 105
  vlan 5
  mcast-group 233.0.0.5
exit
!
!
ip maximum-paths 32
!
hostname Leaf_3
!
interface TenGigabitEthernet1/0/9
  description Host2_mcast
  switchport mode trunk
  switchport trunk allowed vlan add 2-5
  switchport forbidden default-vlan
exit
!
interface TenGigabitEthernet1/0/11
  description Host2
  switchport access vlan 1000
exit
!
interface HundredGigabitEthernet1/0/1
  description Spine_1
```

```

ip address 172.16.3.1 255.255.255.252
ip pim
exit
!
interface HundredGigabitEthernet1/0/2
description Spine_2
ip address 172.16.3.5 255.255.255.252
ip pim
exit
!
interface loopback1
ip address 10.0.0.3 255.255.255.255
ip pim
ip igmp static-group 233.0.0.2
ip igmp static-group 233.0.0.3
ip igmp static-group 233.0.0.4
ip igmp static-group 233.0.0.5
exit
!
!
router ospf 1
network 172.16.3.1 area 0.0.0.0
network 172.16.3.5 area 0.0.0.0
router-id 10.0.0.3
timers spf delay 0
redistribute connected subnets
exit
!
interface ip 172.16.3.1
ip ospf network point-to-point
exit
!
interface ip 172.16.3.5
ip ospf network point-to-point
exit
!
router bgp 65500
bgp router-id 10.0.0.3
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
!
peer-group SPINE_GROUP
remote-as 65500
update-source loopback 1
fall-over bfd
exit
!
neighbor 10.0.1.1
advertisement-interval 1 withdraw 1
peer-group SPINE_GROUP
address-family ipv4 unicast
exit
!
address-family l2vpn evpn
exit
exit
!

```

```
neighbor 10.0.2.2
  advertisement-interval 1 withdraw 1
  peer-group SPINE_GROUP
  address-family ipv4 unicast
  exit
  !
  address-family l2vpn evpn
  exit
  exit
exit
!
!
ip multicast-routing pim
!
ip pim rp-address 10.100.100.100
!
ip multicast multipath group-paths-num
!
end
```

## ТЕХНИЧЕСКАЯ ПОДДЕРЖКА

Для получения технической консультации по вопросам эксплуатации оборудования ООО «Предприятие «ЭЛТЕКС» вы можете обратиться в Сервисный центр компании:

Форма обратной связи на сайте: <https://eltex-co.ru/support/>

Servicedesk: <https://servicedesk.eltex-co.ru>

На официальном сайте компании вы можете найти техническую документацию и программное обеспечение для продукции ООО «Предприятие «ЭЛТЕКС», обратиться к базе знаний или оставить интерактивную заявку:

Официальный сайт компании: <https://eltex-co.ru/>

База знаний: <https://docs.eltex-co.ru/display/EKB/Eltex+Knowledge+Base>

Центр загрузок: <https://eltex-co.ru/support/downloads>