

Легкий пример конфигурации виртуальной сети

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Общие сведения](#)

[Настройка](#)

[Схема сети](#)

[Настройте EVN](#)

[Настройте транк VNET](#)

[Список транка](#)

[Атрибуты транка на VRF](#)

[Метки VNET на ссылку](#)

[Проверка](#)

[Устранение неполадок](#)

[Дополнительные сведения](#)

Введение

Этот документ описывает функцию Легкой виртуальной сети (EVN), которая разработана для обеспечения легкого, просто настраиваемого механизма виртуализации в сетях уровня кампуса. Это усиливает современные технологии, такие как Виртуальная маршрутизация и Облегченный Передачей (Облегченный VRF) и инкапсуляция dot1q, и не представляет нового протокола.

Предварительные условия

Требования

Для этого документа отсутствуют особые требования.

Используемые компоненты

Сведения в документе приведены на основе данных версий аппаратного и программного

обеспечения:

- Cisco Catalyst 6000 коммутаторов Серии (Cat6k), которые работают под управлением ПО версии 15.0 (1) SY1
- Маршрутизаторы Series Aggregated Services Cisco 1000 (ASR1000), которые работают под управлением ПО версии 3.2 с
- Маршрутизаторы интегрированных служб Cisco 3925 и 3945, которые выполняют Cisco IOS® Versions 15.3 (2) T и позже
- Cisco Catalyst 4500 (Cat4500) и 4900 коммутаторов Серии (Cat4900), которые работают под управлением ПО версии 15.1 (1) SG

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Общие сведения

Вот обзор функции EVN:

- Использование функции EVN, Облегченное VRF для создания нескольких (до 32) контексты маршрутизации.
- Подключение в Виртуальной маршрутизации и Передаче (VRF) между устройствами Layer3 обеспечено через Виртуальную сеть (VNET) транки.
- Транки VNET являются обычными транками dot1q.
- Каждый VRF, который должен быть транспортирован через транки VNET, должен быть настроен с меткой VNET.
- Каждая метка VNET равняется метке dot1q.
- **Подинтерфейсы dot1q автоматически созданы и скрыты.**
- **Конфигурация основного интерфейса наследована всеми (скрытыми) подинтерфейсами.**
- Отдельные примеры протоколов маршрутизации должны использоваться в каждом VRF по транкам VNET для объявления префиксной достижимости.
- Динамический маршрут, протекающий между VRF (настроенный против статических маршрутов), позволен без использования Протокола BGP.
- Функция поддерживается для IPv4 и IPv6.

Настройка

Используйте информацию, которая описана в этом разделе для настройки функции EVN.

Примечание: [Чтобы получить подробные сведения о командах в данном документе, используйте Средство поиска команд \(только для зарегистрированных клиентов\).](#)

Схема сети

Эта сетевая установка используется для иллюстрирования конфигурации EVN и команд показа:

Вот некоторые важные замечания об этой настройке:

- Два VRF определены (**CUST-A** и **CUST-B**), которые несут из ядра сети через транки VNET.
- Протокол OSPF используется в VRF для объявления достижимости.
- Общий сервер хостов A COM VRF (**100.1.1.100**), который должен быть достижимым и от CUST-A VRF и от CUST-B.
- Образ, который используется, является **i86bi_linux-adventerprisek9-ms.153-1. S**.

Совет: Cisco IOS на Linux (IOL) настройка, которая используется, доступна [здесь](#).

Настройте EVN

Выполните эти шаги для настройки функции EVN:

1. Настройте VRF definition:

```
vrf definition [name]
vnet tag [2-4094]
!
address-family ipv4|ipv6
exit-address-family
```

! Вот некоторые важные замечания об этой конфигурации:

Cisco рекомендует использовать метки в диапазоне 2 - 1,000. Не используйте зарезервированные VLAN 1,001 - 1,005. Расширенные VLAN 1,006 - 4,094 могут использоваться в случае необходимости.

Метка VNET не должна использоваться текущей VLAN.

Метки VNET должны быть тем же на всех устройствах для любого данного VRF.

Адрес-family ipv4|ipv6 должен быть настроен для активации VRF в связанном AF.

Нет никакой потребности определить Направление маршрута (RD), потому что EVN не использует BGP.

С этой настройкой VRF должны быть определены на всех 4х центральные маршрутизаторы. Например, на БАЗОВОМ 1:

```
vrf definition CUST-A
 vnet tag 100
 !
 address-family ipv4
 exit-address-family
vrf definition CUST-B
 vnet tag 200
 !
 address-family ipv4
```

exit-address-familyИспользование тот же VNET наклеивает все маршрутизаторы для этих VRF. На БАЗОВЫХ 4 COM VRF не требует метки VNET. Цель состоит в том, чтобы поддержать тот VRF локальным на БАЗОВЫХ 4 и настроить утечку и перераспределение для обеспечения доступа к общему серверу от CUST-A и CUST-B.

Введите эту команду для проверки различных счетчиков VNET:

```
CORE-1#show vnet counters
Maximum number of VNETs supported: 32
Current number of VNETs configured: 2
Current number of VNET trunk interfaces: 2
Current number of VNET subinterfaces: 4
Current number of VNET forwarding interfaces: 6
CORE-1#
```

2. Настройте транк VNET:

```
CORE-1#show vnet counters
Maximum number of VNETs supported: 32
Current number of VNETs configured: 2
Current number of VNET trunk interfaces: 2
Current number of VNET subinterfaces: 4
Current number of VNET forwarding interfaces: 6
```

CORE-1#Вот некоторые важные замечания об этой конфигурации:

Команда **транка vnet** создает столько же подинтерфейсов dot1q сколько количество VRF, которые определены с меткой VNET.

Команда **транка vnet** не может сосуществовать с некоторыми вручную настроенными подинтерфейсами на том же физическом интерфейсе.

Эта конфигурация позволена на маршрутизируемых интерфейсах (не порты коммутатора), физическая и portchannel.

IP-адреса (и другие команды), которые применены на физический интерфейс, наследованы подинтерфейсами.

Подинтерфейсы для всех VRF используют тот же IP-адрес.

С этой настройкой существует два VRF VNET, таким образом, два подинтерфейса автоматически созданы на интерфейсе, который настроен как транк VNET. Можно ввести **команду show derived-config** для просмотра скрытой конфигурации, которая автоматически создана:

Вот конфигурация, которая в настоящее время выполняется:

```
CORE-1#show run | s Ethernet0/0
interface Ethernet0/0
  vnet trunk
  ip address 192.168.1.1 255.255.255.252
  !
```

CORE-1#Вот полученная конфигурация:

```
CORE-1#show derived-config | s Ethernet0/0
interface Ethernet0/0
  vnet trunk
  ip address 192.168.1.1 255.255.255.252
Interface Ethernet0/0.100
  description Subinterface for VNET CUST-A
  encapsulation dot1Q 100
  vrf forwarding CUST-A
  ip address 192.168.1.1 255.255.255.252
interface Ethernet0/0.200
  description Subinterface for VNET CUST-B
  encapsulation dot1Q 200
  vrf forwarding CUST-B
  ip address 192.168.1.1 255.255.255.252
```

CORE-1#Как показано все подинтерфейсы наследовали IP-адрес основного интерфейса.

3. Назначьте край (sub) интерфейсы к VRF. Для присвоения интерфейса или подинтерфейса к VRF VNET, используйте ту же процедуру в качестве используемого для присвоения VRF обычно:

```
CORE-1#show derived-config | s Ethernet0/0
interface Ethernet0/0
  vnet trunk
  ip address 192.168.1.1 255.255.255.252
Interface Ethernet0/0.100
  description Subinterface for VNET CUST-A
  encapsulation dot1Q 100
  vrf forwarding CUST-A
  ip address 192.168.1.1 255.255.255.252
interface Ethernet0/0.200
  description Subinterface for VNET CUST-B
  encapsulation dot1Q 200
  vrf forwarding CUST-B
  ip address 192.168.1.1 255.255.255.252
```

CORE-1#С этой настройкой конфигурация применена на БАЗОВЫЙ 1 и БАЗОВЫЕ 4. Вот пример для БАЗОВЫХ 4:

```
CORE-1#show derived-config | s Ethernet0/0
interface Ethernet0/0
  vnet trunk
  ip address 192.168.1.1 255.255.255.252
Interface Ethernet0/0.100
  description Subinterface for VNET CUST-A
  encapsulation dot1Q 100
  vrf forwarding CUST-A
  ip address 192.168.1.1 255.255.255.252
interface Ethernet0/0.200
  description Subinterface for VNET CUST-B
  encapsulation dot1Q 200
  vrf forwarding CUST-B
  ip address 192.168.1.1 255.255.255.252
CORE-1#
```

4. Настройте протоколы маршрутизации для каждого VRF (это не является определенным для EVN или VNET):

```
router ospf x vrf [name]
network x.x.x.x y.y.y.y area x
```

... **Примечание:** Эта конфигурация должна включать адреса транка VNET, а также адреса краевого интерфейса.

С этой настройкой два процесса OSPF определены, один на VRF:

```
CORE-1#show run | s router os
router ospf 1 vrf CUST-A
network 10.1.1.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
router ospf 2 vrf CUST-B
network 10.2.1.0 0.0.0.255 area 0
network 192.168.1.0 0.0.0.255 area 0
```

CORE-1#**Можно перейти в режим контекста маршрутизации для просмотра информации, которая отнесена к определенному VRF без спецификаций VRF в каждой команде:**

```
CORE-1#routing-context vrf CUST-A
CORE-1%CUST-A#
CORE-1%CUST-A#show ip protocols
*** IP Routing is NSF aware ***
Routing Protocol is "ospf 1"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Router ID 192.168.1.13
  It is an area border router
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
  Maximum path: 4
  Routing for Networks:
    10.1.1.0 0.0.0.255 area 0
    192.168.1.0 0.0.0.255 area 0
  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.1.9      110          1d00h
    192.168.1.14     110          1d00h
  Distance: (default is 110)
CORE-1%CUST-A#
CORE-1%CUST-A#show ip os neighbor
Neighbor ID      Pri  State           Dead Time   Address      Interface
192.168.1.14     1    FULL/DR         00:00:30   192.168.1.14 Ethernet1/0.100
192.168.1.5      1    FULL/BDR        00:00:37   192.168.1.2  Ethernet0/0.100
10.1.1.2         1    FULL/BDR        00:00:33   10.1.1.2     Ethernet2/0
```

CORE-1%CUST-A#**Примечание:** Команда **show ip protocols** вывела показы только информация, которая отнесена к выбранному VRF.

При просмотре Routing Information Base (RIB) для обоих VRF можно проверить удаленную подсеть через два транка VNET:

```
CORE-1%CUST-A#show ip route 10.1.2.0
Routing Table: CUST-A
Routing entry for 10.1.2.0/24
  Known via "ospf 1", distance 110, metric 30, type intra area
  Last update from 192.168.1.2 on Ethernet0/0.100, 1d00h ago
  Routing Descriptor Blocks:
  * 192.168.1.14, from 192.168.1.9, 1d00h ago, via Ethernet1/0.100
    Route metric is 30, traffic share count is 1
    192.168.1.2, from 192.168.1.9, 1d00h ago, via Ethernet0/0.100
      Route metric is 30, traffic share count is 1
CORE-1%CUST-A#
CORE-1%CUST-A#routing-context vrf CUST-B
CORE-1%CUST-B#
CORE-1%CUST-B#show ip route 10.2.2.0
Routing Table: CUST-B
Routing entry for 10.2.2.0/24
  Known via "ospf 2", distance 110, metric 30, type intra area
  Last update from 192.168.1.2 on Ethernet0/0.200, 1d00h ago
```

Routing Descriptor Blocks:

```
* 192.168.1.14, from 192.168.1.6, 1d00h ago, via Ethernet1/0.200
  Route metric is 30, traffic share count is 1
  192.168.1.2, from 192.168.1.6, 1d00h ago, via Ethernet0/0.200
  Route metric is 30, traffic share count is 1
```

CORE-1%**CUST-B**#

CORE-1%**CUST-B**#exit

CORE-1#

CORE-1#

5. Определите перераспределение маршрутов между VRF. Перераспределение маршрутов выполнено через репликацию маршрута. Например, некоторые маршруты в VRF могли бы быть сделаны доступными для другого VRF:

```
vrf definition VRF-X
  address-family ipv4|ipv6
  route-replicate from vrf VRF-Y unicast|multicast
```

[**route-origin**] [**route-map** [name]] Вот некоторые важные замечания об этой конфигурации:

RIB для **VRF-X** имеет доступ к выбранным маршрутам, на основе параметров командной строки от **VRF-Y**.

Реплицированные маршруты в **VRF-X** отмечены [**+**] флаг.

Опция **групповой адресации** позволяет использование маршрутов от другого VRF для Пересылки по обратному пути (RPF).

Источник маршрута может иметь одно из этих значений:

всеbgrpподключеноeIGRPIs-ISмобильныйodroSPFripСтатичный

В отличие от названия указывает, маршруты не реплицированы или дублированы; дело обстоит так с обычной утечкой через BGP общий RT, который не использует дополнительную память.

С этой настройкой перераспределение маршрутов используется на БАЗОВЫХ 4 для обеспечения доступа от CUST-A и CUST-B к COM (и наоборот):

```
vrf definition CUST-A
address-family ipv4
route-replicate from vrf COM unicast connected
!
vrf definition CUST-B
address-family ipv4
route-replicate from vrf COM unicast connected
!
vrf definition COM
address-family ipv4
route-replicate from vrf CUST-A unicast ospf 1 route-map USERS
route-replicate from vrf CUST-B unicast ospf 2 route-map USERS
!
route-map USERS permit 10
match ip address prefix-list USER-SUBNETS
!
ip prefix-list USER-SUBNETS seq 5 permit 10.0.0.0/8 le 32
```

CORE-4#**show ip route vrf CUST-A**

Routing Table: COM

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP

```

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area,* - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
a - application route
+ - replicated route, % - next hop override

```

```

...
10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O   10.1.1.0/24 [110/30] via 192.168.1.10, 3d19h, Ethernet1/0.100
    [110/30] via 192.168.1.5, 3d19h, Ethernet0/0.100
100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C + 100.1.1.0/24 is directly connected (COM), Ethernet4/0

```

```

CORE-4#show ip route vrf CUST-B
... 10.0.0.0/8 is variably subnetted, 3 subnets, 2 masks
O   10.2.1.0/24 [110/30] via 192.168.1.10, 1d00h, Ethernet1/0.200
    [110/30] via 192.168.1.5, 1d00h, Ethernet0/0.200
100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C + 100.1.1.0/24 is directly connected (COM), Ethernet4/0

```

```

CORE-4#show ip route vrf COM
...
10.0.0.0/24 is subnetted, 2 subnets
O + 10.1.1.0 [110/30] via 192.168.1.10 (CUST-A), 3d19h, Ethernet1/0.100
    [110/30] via 192.168.1.5 (CUST-A), 3d19h, Ethernet0/0.100
O + 10.2.1.0 [110/30] via 192.168.1.10 (CUST-B), 1d00h, Ethernet1/0.200
    [110/30] via 192.168.1.5 (CUST-B), 1d00h, Ethernet0/0.200
100.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C   100.1.1.0/24 is directly connected, Ethernet4/0

```

На этом этапе реплицированные маршруты не распространяются в Протоколе IGP, поэтому только CE, 2 и CE-B-2 имеют доступ к сервису COM (100.1.1.100), не CE 1 и B-1 CE.

Можно также использовать перераспределение маршрутов от или до глобальной таблицы:

```

vrf definition VRF-X
  address-family ipv4
  route-replicate from vrf >global unicast|multicast [route-origin]
[route-map [name]]
  exit-address-family
!
exit
!
global-address-family ipv4 unicast
  route-replicate from vrf [vrf-name] unicast|multicast [route-origin]
[route-map [name]]

```

6. Определите распространение перераспределения маршрутов. Пропущенные маршруты не дублированы в целевом RIB VRF. Другими словами, они не часть целевого RIB VRF. Обычное перераспределение между процессами маршрутизатора не работает, таким образом, необходимо явно определить соединение VRF RIB, которому принадлежит маршрут:

```

router ospf x vrf VRF-X
  redistribute vrf VRF-Y [route-origin] [route-map [name]]

```

Пропущенные маршруты от VRF-Y перераспределены в процессе OSPF, который выполняется в VRF-X. Вот пример на БАЗОВЫХ 4:

```

router ospf x vrf VRF-X
  redistribute vrf VRF-Y [route-origin] [route-map [name]]

```

Route-map не необходим в этом

случае, так как существует только один связанный маршрут в COM VRF. Существует теперь достижимость к сервису COM (100.1.1.100) от CE 1 и B-1 CE:

```
CE-A-1#ping 100.1.1.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.1.100, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
CE-A-1#
```

```
CE-B-1#ping 100.1.1.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 100.1.1.100, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
CE-B-1#
```

Настройте транк VNET

Этот раздел предоставляет сведения, который можно использовать для настройки транка VNET.

Список транка

По умолчанию все VRF, которые настроены с меткой VNET, позволены на всех транках VNET. Список транка позволяет вам задавать список санкционированных VRF на транке VNET:

```
vrf list [list-name]
  member [vrf-name]
!
interface GigabitEthernetx/x
  vnet trunk list [list-name]
```

Примечание: Должна быть одна линия на позволенный VRF.

Как пример, БАЗОВЫЙ 1 настроен для CUST-B VRF на транке VNET между БАЗОВЫМ 1 и Core 2:

```
vrf list [list-name]
  member [vrf-name]
!
interface GigabitEthernetx/x
  vnet trunk list [list-name]
```

Как показано OSPF, взаимодействующий для CUST-B VRF через транк, выключается:

```
%OSPF-5-ADJCHG: Process 2, Nbr 192.168.1.2 on Ethernet0/0.200 from FULL to DOWN,
Neighbor Down: Interface down or detached
```

Подинтерфейс для CUST-B VRF удален:

```
CORE-1#show derived-config | b Ethernet0/0
interface Ethernet0/0
  vnet trunk list TEST
  ip address 192.168.1.1 255.255.255.252
!
interface Ethernet0/0.100
  description Subinterface for VNET CUST-A
```

```
encapsulation dot1q 100
vrf forwarding CUST-A
ip address 192.168.1.1 255.255.255.252
!
```

Атрибуты транка на VRF

По умолчанию подинтерфейсы dot1q наследовали параметры физического интерфейса так, чтобы подинтерфейсы для всех VRF имели те же атрибуты (такой, как стоит и аутентификация). Можно настроить параметры магистрали на метку VNET:

```
CORE-1#show derived-config | b Ethernet0/0
interface Ethernet0/0
 vnet trunk list TEST
 ip address 192.168.1.1 255.255.255.252
!
interface Ethernet0/0.100
 description Subinterface for VNET CUST-A
 encapsulation dot1q 100
 vrf forwarding CUST-A
 ip address 192.168.1.1 255.255.255.252
!
```

Можно настроить эти параметры:

```
CORE-1(config-if-vnet)#?
Interface VNET instance override configuration commands:
 bandwidth      Set bandwidth informational parameter
 default         Set a command to its defaults
 delay         Specify interface throughput delay
 exit-if-vnet   Exit from VNET submode
 ip           Interface VNET submode Internet Protocol config commands
 no             Negate a command or set its defaults
 vnet          Configure protocol-independent VNET interface options
CORE-1(config-if-vnet)#
CORE-1(config-if-vnet)#ip ?
 authentication      authentication subcommands
 bandwidth-percent   Set EIGRP bandwidth limit
 dampening-change    Percent interface metric must change to cause update
 dampening-interval  Time in seconds to check interface metrics
 hello-interval      Configures EIGRP-IPv4 hello interval
 hold-time           Configures EIGRP-IPv4 hold time
 igmp                IGMP interface commands
 mfib                Interface Specific MFIB Control
 multicast            IP multicast interface commands
 next-hop-self       Configures EIGRP-IPv4 next-hop-self
 ospf                OSPF interface commands
 pim                 PIM interface commands
 split-horizon       Perform split horizon
 summary-address     Perform address summarization
 verify              Enable per packet validation
CORE-1(config-if-vnet)#ip
```

В данном примере изменена стоимость OSPF на VRF для БАЗОВОГО 1, таким образом, путь Core 2 используется для CUST-A, и БАЗОВЫЕ 3 пути для CUST-B (стоимость по умолчанию равняется 10):

```
interface Ethernet0/0
vnet name CUST-A
ip ospf cost 8
!
vnet name CUST-B
```

```
ip ospf cost 12
!
```

```
CORE-1#show ip route vrf CUST-A 10.1.2.0
```

```
Routing Table: CUST-A
Routing entry for 10.1.2.0/24
Known via "ospf 1", distance 110, metric 28, type intra area
Last update from 192.168.1.2 on Ethernet0/0.100, 00:05:42 ago
Routing Descriptor Blocks:
* 192.168.1.2, from 192.168.1.9, 00:05:42 ago, via Ethernet0/0.100
Route metric is 28, traffic share count is 1
CORE-1#
CORE-1#show ip route vrf CUST-B 10.2.2.0
```

```
Routing Table: CUST-B
Routing entry for 10.2.2.0/24
Known via "ospf 2", distance 110, metric 30, type intra area
Last update from 192.168.1.14 on Ethernet1/0.200, 00:07:03 ago
Routing Descriptor Blocks:
* 192.168.1.14, from 192.168.1.6, 1d18h ago, via Ethernet1/0.200
Route metric is 30, traffic share count is 1
CORE-1#
```

Метки VNET на ссылку

По умолчанию метка VNET, которая определена в VRF definition, используется для всех транков. Однако можно использовать другую метку VNET на транк.

Данный пример описывает сценарий, где вы связаны с non-EVN устройством с поддержкой, и вы используете Облегченный VRF с ручным транком, и глобальная метка VNET используется другой VLAN:

С этой настройкой метка VNET, которая используется на транке между БАЗОВЫМ 1 и Core 2 для CUST-A, изменена от 100 до 101:

```
interface Ethernet0/0
vnet name CUST-A
vnet tag 101
```

После того, как это изменение происходит на БАЗОВОМ 1, новый подинтерфейс создан:

```
CORE-1#show derived-config | b Ethernet0/0
interface Ethernet0/0
vnet trunk
ip address 192.168.1.1 255.255.255.252
!
interface Ethernet0/0.101
description Subinterface for VNET CUST-A
encapsulation dot1q 101
vrf forwarding CUST-A
ip address 192.168.1.1 255.255.255.252
!
interface Ethernet0/0.200
description Subinterface for VNET CUST-B
encapsulation dot1q 200
vrf forwarding CUST-B
ip address 192.168.1.1 255.255.255.252
```

Если это изменение происходит только на одном конце, то подключение потеряно в связанном VRF, и OSPF выключается:

```
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.1.5 on Ethernet0/0.101 from FULL to DOWN,  
Neighbor Down: Dead timer expired
```

Как только та же метка VNET используется на Core 2, подключение восстановлено, и dot1q помечают **101**, используется на том транке, в то время как **100** все еще используется на БАЗОВОМ 1 к БАЗОВЫМ 3 транкам:

```
%OSPF-5-ADJCHG: Process 1, Nbr 192.168.1.5 on Ethernet0/0.101 from LOADING to  
FULL, Loading Done
```

Проверка

В настоящее время для этой конфигурации нет процедуры проверки.

Устранение неполадок

Для этой конфигурации в настоящее время нет сведений об устранении проблем.

Дополнительные сведения

- [Легкая виртуальная сеть - упрощение виртуализации сети уровня 3](#)
- [Cisco Systems – техническая поддержка и документация](#)