

Использование vemlog для отладки Nexus 1000v проблемы

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Рабочие команды vemlog](#)

[Уровни отладки vemlog](#)

[Шаг 1: Очистка Текущих параметров настройки Перехвата](#)

[Шаг 2: Установка vemlog перехватывает параметры](#)

[Шаг 3: Просмотр выходных данных](#)

[Шаг 4. : Восстановление vemlog к обычным уровням регистрации](#)

[Дополнительные сведения](#)

Введение

Часть Действительного модуля Ethernet (VEM) Nexus 1000v имеет встроенные **команды отладки**, которые можно использовать для помощи в устранении проблем. Эти команды vemlog позволяют вам понимание тому, что делают определенные процессы VEM, какие команды они отсылают, и какой ответ они получают. Например, вы могли бы использовать их в этих ситуациях:

- Port-channel LACP не подходит между Nexus 1000v и восходящим коммутатором. Вы использовали бы комбинацию "vemlog" на хосте и "отладки" на восходящем коммутаторе для сужения проблемы.
- QoS не работает правильно, и вы хотите проверить, что VEM устанавливает значения правильно.
- Вы хотите видеть изменения к связи VSM-VEM.

Случай наиболее популярного способа использования для того, чтобы выполнить эти команды отлаживает LACP. Использование модуля отладки "sflacp" и "debug lacp" на восходящем коммутаторе, который можно обнаружить, какое устройство неправильно функционирует, или если ссылка между этими двумя устройствами отбрасывает BPDU LACP.

Уровни по умолчанию обычно имеют достаточно информации, таким образом, журнал может быть собран и проанализирован, не включая дополнительные отладки.

[Предварительные условия](#)

Требования

Для этого документа отсутствуют особые требования.

Используемые компоненты

Настоящий документ не имеет жесткой привязки к каким-либо конкретным версиям программного обеспечения и оборудования.

Условные обозначения

[Дополнительные сведения об условных обозначениях см. в документе Условные обозначения технических терминов Cisco.](#)

Рабочие команды vemlog

При наличии каких-либо вопросов о синтаксисе этих команд, можно ввести “vemlog” в CLI ESXi и получить выходные данные справки. Нет никакой завершённой вкладки или ‘?’ для этих команд. Необходимо полностью вывести их, или они не будут работать правильно. Для выполнения команд **vemlog**, вам нужно одно из придерживающегося:

- Консольный доступ SSH / локальный консольный доступ к рассматриваемому хосту ESXi. Доступ SSH предпочтителен, потому что можно легко перейти резервное копирование.
- Консольный доступ SSH / локальный консольный доступ к VSM и модулю должны быть активными в VSM. Еще раз доступ SSH предпочтителен. При использовании этого метода, необходимо будет снабдить каждую команду предисловием с **модулем vem #, выполняются**, где # является номером модуля VEM.

Уровни отладки vemlog

Vemlog на VEM подобен системному журналу на обычном коммутаторе. Как обычный коммутатор, существуют уровни важности:

```
e - Error
n - Notification
w - Warning
i - Information
d - Debug
p - Print
t - Temporary
```

Это оценивается от самого серьезного до наименьшего количества (с менее серьезными сообщениями, выводящими далекие дополнительные сведения). По умолчанию vemlog установлен в ENW (Ошибка, Уведомление, Предупреждение).

Шаг 1: Очистка Текущих параметров настройки Перехвата

Для устранения проблем отдельной проблемы ценно отключить перехваты по умолчанию.

```
vemlog stop
vemlog clear
vemlog debug all none
```

Эти команды остановят любые текущие команды **vemlog**, которые происходят и перезагрузят vemlog для очищения параметров настройки перехвата. Можно проверить это со следующей командой:

```
~ #
~ #
~ # vemlog show debug
      Module                Available                Printing
      all                   ENWID P ( 95)          ( 0)
      sfheapshow            ENWID P ( 95)          ( 0)
      sfheap                ENWID P ( 95)          ( 0)
      sfmempoolshow         ENWID P ( 95)          ( 0)
      sfmempool             ENWID P ( 95)          ( 0)
      if_bridge_rt         ENWID P ( 95)          ( 0)
      sfint32list           ENWID P ( 95)          ( 0)
      sflist                ENWID P ( 95)          ( 0)
      cbridge_proc         ENWID P ( 95)          ( 0)
      vssnet                ENWID P ( 95)          ( 0)
      sfpacket              ENWID P ( 95)          ( 0)
      sfbase                ENWID P ( 95)          ( 0)
      sfschedlog           ENWID P ( 95)          ( 0)
      sfsched              ENWID P ( 95)          ( 0)
      sfbid                 ENWID P ( 95)          ( 0)
```

Как вы видите здесь, команда **vemlog show debug** отобразит текущие параметры настройки отладки. По умолчанию для всех модулей для “Печати” 0, или не печать.

[Шаг 2: Установка vemlog перехватывает параметры](#)

Теперь, когда у вас есть пустой список для начала с, пора заставить соookie перехватывать. Для данного примера перехвачены общие журналы порта:

```
~ #
~ #
~ # vemlog show debug | grep sfport
      sfport                ENWID P ( 95)          ( 0)
      sfporttable           ENWID P ( 95)          ( 0)
      sfportchannel         ENWID P ( 95)          ( 0)
      sfportnotify         ENWID P ( 95)          ( 0)
~ # vemlog debug sfport debug
~ # vemlog show debug | grep sfport
      sfport                ENWID P ( 95)          D ( 16)
      sfporttable           ENWID P ( 95)          ( 0)
      sfportchannel         ENWID P ( 95)          ( 0)
      sfportnotify         ENWID P ( 95)          ( 0)
~ #
```

Вторая команда там имеет этот синтаксис:

```
vemlog debug [module|all] [-][all|none|default|e|w|n|i|d|p|t]
```

Где “debug_module” является пакетом vemlog, вы хотите перехватить. Обычно “отладка” достаточно для устранения проблем.

Если вы добавляете [-] к команде, что уровень отладки будет удален для того модуля. Этот снимок экрана имеет примеры этого:

```
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      ENW      L (135)
~ # vemlog debug sfvlan -w
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      EN       L (131)
~ # vemlog debug sfvlan d
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      EN D     L (147)
~ # vemlog debug sfvlan all
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      ENWIDTPL (255)
~ # vemlog debug sfvlan default
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      ENW      L (135)
~ # vemlog debug sfvlan none
~ # vemlog show debug | grep sfvlan
svvlan          ENWID PL (223)      ( 0)
```

Можно посмотреть все доступные модули отладки с командой `vemlog show debug`.

Как только вы задали, которые пакеты контролировать, затем запустите `vemlog` с команды `vemlog start`:

```
~ #
~ # vemlog start
Started log
~ #
```

Шаг 3: Просмотр выходных данных

```
~ #
~ # vemlog stop
Suspended log
~ # vemlog show all
Timestamp      Entry CPU  Mod Lv  Message
May 28 23:41:06.857619 0 2 1 16 Debug sf_parse_n_check_packet nh 0x0x4100061c6056 encsize 22 enctype 4 rx_type 0
May 28 23:41:06.857632 1 2 1 16 Debug Egress 9 packets 1 bytes 593
May 28 23:41:06.857634 2 2 1 16 Debug Egress sending packet list ltl 9 packets 1
May 28 23:41:06.857663 3 2 1 16 Debug Egress free packet list time 81158
May 28 23:41:06.857702 4 4 1 16 Debug sf_parse_n_check_packet nh 0x0x4100062e1157 encsize 22 enctype 4 rx_type 0
May 28 23:41:06.857710 5 4 1 16 Debug po_vlan 224 native 1
May 28 23:41:06.857712 6 4 1 16 Debug Egress 18 packets 1 bytes 640
May 28 23:41:06.857713 7 4 1 16 Debug Egress sending packet list ltl 18 packets 1
May 28 23:41:06.857714 8 4 1 16 Debug Tag packet 0x41000633e500
May 28 23:41:06.857725 9 4 1 16 Debug Egress free packet list time 31790
May 28 23:41:11.683412 10 2 1 16 Debug sf_parse_n_check_packet nh 0x0x410006147756 encsize 22 enctype 4 rx_type 0
May 28 23:41:11.683425 11 2 1 16 Debug Egress 9 packets 1 bytes 593
May 28 23:41:11.683427 12 2 1 16 Debug Egress sending packet list ltl 9 packets 1
May 28 23:41:11.683453 13 2 1 16 Debug Egress free packet list time 74829
May 28 23:41:11.683493 14 4 1 16 Debug sf_parse_n_check_packet nh 0x0x4100061cf457 encsize 22 enctype 4 rx_type 0
May 28 23:41:11.683501 15 4 1 16 Debug po_vlan 224 native 1
May 28 23:41:11.683503 16 4 1 16 Debug Egress 18 packets 1 bytes 640
```

Как только вы готовы посмотреть журнал, выполнить команду `vemlog show all`. Это

распечатывает все записи журнала к консоли (или Сеанс SSH). Если вы напрямую подключаетесь к хосту, можно также передать это к файлу так, чтобы можно было скопировать его прочь и проанализировать его позже:

```
~ #  
~ # vemlog show all > /tmp/vemlog  
~ # ls /tmp  
dpafifo  scratch  vemlog  
~ #
```

При перемещении этого файла в `vmfs/volumes / [datastore_name]` каталог, можно переместить его от хоста с помощью браузера хранилища данных в ESXi.

Vemlog также сохранен в журнале vmkernel.

Шаг 4. : Восстановление vemlog к обычным уровням регистрации

```
~ # vemlog stop  
Suspended log  
~ # vemlog clear  
Cleared log  
~ # vemlog debug all default  
~ # vemlog start  
Started log
```

Эти команды восстановят vemlog к стандартной конфигурации. Это важно, чтобы гарантировать, что информация об отладке перехвачена должным образом для дальнейших проблем.

Поскольку vemlog сохранен и в журналах vmkernel и в буфере vemlog, особенно болтливая отладка может лавинно разослать оба из них с ненужной информацией, особенно если их оставляют включенными.

Дополнительные сведения

- [Cisco Systems – техническая поддержка и документация](#)