

Nexus 1000v 問題をデバッグする vemlog の使用

目次

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[vemlog コマンドの実行](#)

[Vemlog デバッグレベル](#)

[ステップ 1: 現在のキャプチャ設定をクリアする方法](#)

[ステップ 2: vemlog キャプチャ パラメータの設定](#)

[ステップ 3: 出力の表示](#)

[ステップ 4: 正常なログ レベルへの vemlog を復元する](#)

[関連情報](#)

概要

Nexus 1000v の仮想 な イーサネットモジュール (VEM) 部分にトラブルシューティングで助けるのに使用できる組み込み debug コマンドがあります。これらの vemlog コマンドはどんなへのコマンドを送信し、どのような応答を受け取っているか、VEM のどんなある特定のプロセスがしているか把握を与えます。たとえば、それらをこの場合使用するかもしれません:

- LACP port-channel は Nexus 1000v とアップストリーム スイッチの間でアップしていません。問題を狭めるのにアップストリーム スイッチのホストおよび「デバッグ」の「vemlog」の組み合わせを利用します。
- QoS は正しくはたらかなくて、VEM が値を正しく設定していることを確認したいと思います。
- VSM-VEM 通信への変更を見たいと思います。

これらのコマンドを実行するためのもっとも一般的な 使用例は LACP をデバッグしています。アップストリーム スイッチのデバッグ モジュール「sflacp」および「デバッグ lacp を使用する」どのデバイスが故障している、または 2 つのデバイス間のリンクが LACP BPDU を廃棄すればか検出することができます。

デフォルトレベルに通常十分な情報があります従ってログは追加デバッグをつけないで集められ、分析することができます。

前提条件

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントは、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

表記法

ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

vemlog コマンドの実行

これらのコマンドの構文についての質問があれば ESXi CLI の「vemlog」を入力し、ヘルプ出力を表示されることができます。あります完全な No タブがまたは「か」。これらのコマンドのため。十分にそれらを入力して下さいか、または正しくはたつきません。vemlog を実行するために次のいずれかを必要とするように命じます:

- 疑わしい ESXi ホストへの SSH/local コンソールアクセス。SSH アクセスは容易にバックアップをスクロールできるので望ましいです。
- VSM およびモジュールへの SSH/local コンソールアクセスは VSM でアクティブでなければなりません。再度 SSH アクセスは望ましいです。#実行するモジュール vem と各コマンドに前書きする必要があるこの方式を使用していれば、#VEM のモジュール番号があるところ。

Vemlog デバッグレベル

VEM の Vemlog は正常なスイッチの syslog に類似したです。正常なスイッチのように、重大度があります:

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

これは最も厳しいから最少へのランク付けされます (ずっと詳細を出力するより少なく厳しいメッセージと)。デフォルトで vemlog は ENW (警告するエラー、通知) に設定されます。

ステップ 1: 現在のキャプチャ設定をクリアする方法

個別の問題点をデフォルト キャプチャをディセーブルにするために解決するためにそれは貴重です。

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

これらのコマンドは起こっている、ためにリセットしましたキャプチャ設定をブランクにするために vemlog を停止し現在の vemlog コマンドを。次のコマンドでこれを確認できます:

ここに見るように、`vemlog show debug` コマンドは現在のデバッグ設定を表示する。「印刷」のすべてのモジュールのためのデフォルトは 0、または印刷です。

ステップ 2: vemlog キャプチャパラメータの設定

から開始するべきまっさらな状態があるのでそれはキャプチャするためにクッキーを設定する時間です。この例に関しては、一般のポート ログはキャプチャされます:

そこの第 2 コマンドにこの構文があります:

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

「debug_module」が vemlog パッケージである一方、キャプチャしたいと思います。通常「デバッグ」はトラブルシューティングのため十分です。

コマンドに[-デバッグ レベルはそのモジュールのために取除かれること]追加すれば。このスクリーンショットにこれの例があります:

`vemlog show debug` コマンドですべての利用可能なデバッグ モジュールを表示できます。

監視するために実装する規定したらそして `vemlog start` コマンドで vemlog を開始して下さい:

ステップ 3: 出力の表示

ログを調べて準備ができていたら `vemlog` を示しますすべてのコマンドを発行して下さい。これはコンソール印刷します (または SSH セッション) にすべての Log エントリを。ホストに直接接続されれば以降それをコピーし、分析できるようにまたファイルにこれを送信できます:

`/vmfs/volumes/ [datastore_name]`ディレクトリにこのファイルを移動すれば ESXi のデータストアブラウザを使用してホストを離れてそれを移動できます。

vemlog はまた vmkernel ログで保存されます。

ステップ 4: 正常なログレベルへの vemlog を復元する

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

これらのコマンドは標準の設定に vemlog を復元する。これはデバッグ情報が未来の問題のためにきちんとキャプチャされるようにして重要です。

vemlog が vmkernel ログおよび vemlog のバッファで両方保存されるので、特に話好きなデバッグは不必要な情報と特にイネーブルのままになれば両方にあふれることができます。

関連情報

- [テクニカルサポートとドキュメント - Cisco Systems](#)