Nexus 1000v 問題をデバッグする vemlog の使用

内容

[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のクリア](#)
[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：vemlogを通常の口グレベルに復元する](#)
[関連情報](#)

概要

Nexus 1000vの仮想イーサネットモジュール(VEM)部分には、トラブルシューティングに役立つ**debugコマンドが組み込**まれています。これらのvemlogコマンドを使用すると、VEMの特定のプロセス、送信しているコマンド、および受信している応答の種類を把握できます。たとえば、次のような状況で使用できます。

- Nexus 1000vとアップストリームスイッチの間でLACPポートチャネルがアップ状態にならない。問題を絞り込むには、ホストで「vemlog」、アップストリームスイッチで「debug」の組み合わせを使用します。
- QoSが正しく動作していないため、VEMが値を正しく設定していることを確認する必要があります。
- VSM-VEM通信の変更を確認する必要があります。

これらのコマンドを実行する際の最も一般的な使用例は、LACPのデバッグです。アップストリームスイッチでデバッグモジュール「sflacp」と「debug lacp」を使用すると、どのデバイスが誤動作しているか、2つのデバイス間のリンクでLACP BPDUが廃棄されているかどうかを検出できます。

通常、デフォルトレベルには十分な情報があるため、追加のデバッグをオンにせずに口グを収集および分析できます。

前提条件

要件

このドキュメントに特有の要件はありません。

使用するコンポーネント

このドキュメントの内容は、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関する質問がある場合は、ESXi CLIで「vemlog」と入力すると、ヘルプ出力が表示されます。タブが完了していないか、'?'がありません。 を参照してください。完全に入力する必要があります。入力しないと正しく動作しません。vemlogコマンドを実行するには、次のいずれかが必要です。

- 対象のESXiホストへのSSH/ローカルコンソールアクセス。簡単にスクロールバックできるので、SSHアクセスが推奨されます。
- VSMへのSSH/ローカルコンソールアクセスとモジュールがVSMでアクティブである必要があります。もう一度、SSHアクセスが優先されます。この方法を使用する場合は、すべてのコマンドの先頭に**module vem # executeを付ける必要があります。ここで#はVEMのモジュール番号です。**

Vemlogデバッグレベル

VEMのVemlogは、通常のスイッチのsyslogに似ています。通常のスイッチと同様に、重大度レベルがあります。

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

これは、最も重大なものから最も低いものにランクされます（メッセージの重大度が低いほど、より多くの情報が出力されます）。 デフォルトでは、vemlogはENW（エラー、通知、警告）に設定されています。

ステップ 1：現在のキャプチャ設定のクリア

特定の問題のトラブルシューティングでは、デフォルトキャプチャを無効にすることが重要です。

```
vemlog stop
vemlog clear
```

```
vemlog debug all none
```

これらのコマンドは、現在発生しているvemlogコマンドをすべて停止し、vemlogを空のキャプチャ設定にリセットします。これを確認するには、次のコマンドを使用します。

ここに示すように、vemlog show debugコマンドは現在のデバッグ設定を表示します。「印刷」のすべてのモジュールのデフォルトは0です。印刷しない場合はデフォルトです。

ステップ 2：vemlogキャプチャパラメータの設定

これで最初にクリーンなスレートが得られたので、Cookieをキャプチャするように設定します。この例では、一般的なポートログがキャプチャされます。

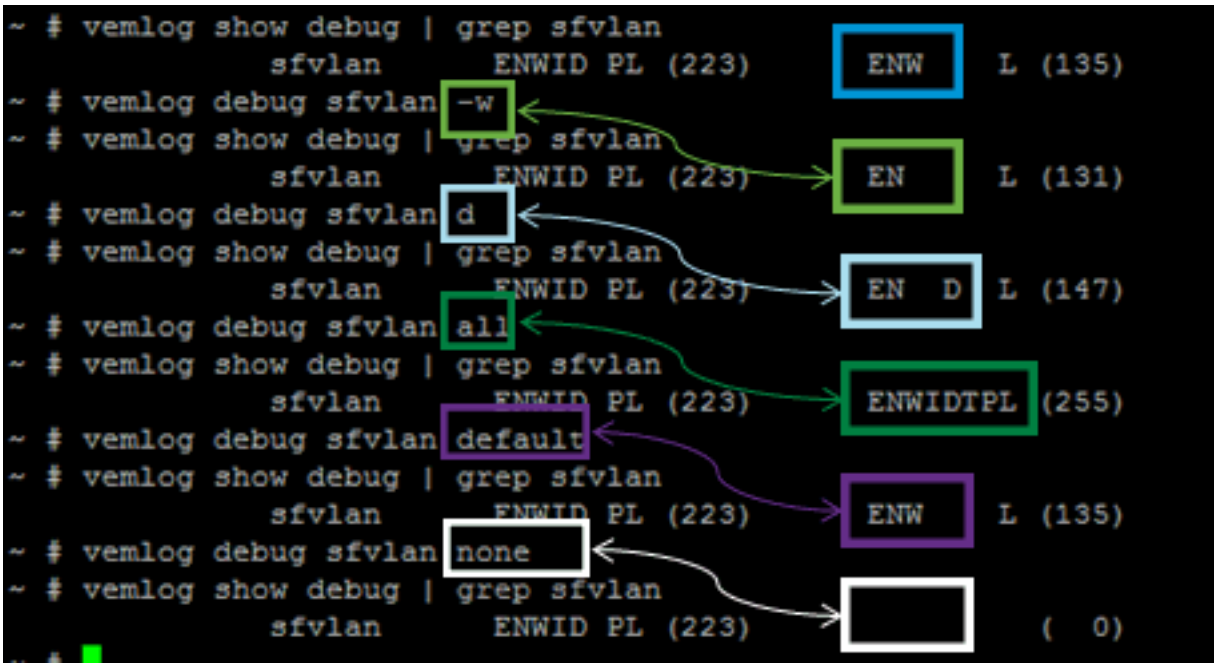
```
~ #
~ #
~ # vemlog show debug | grep sfport
sfport ENWID P ( 95) ( 0)
sfporttable ENWID P ( 95) ( 0)
sfportchannel ENWID P ( 95) ( 0)
sfportnotify ENWID P ( 95) ( 0)
~ # vemlog debug sfport debug
~ # vemlog show debug | grep sfport
sfport ENWID P ( 95) D ( 16)
sfporttable ENWID P ( 95) ( 0)
sfportchannel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

2番目のコマンドの構文は次のとおりです。

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

ここで、「debug_module」は、キャプチャするvemlogパッケージです。通常、「デバッグ」はトラブルシューティングに十分です。

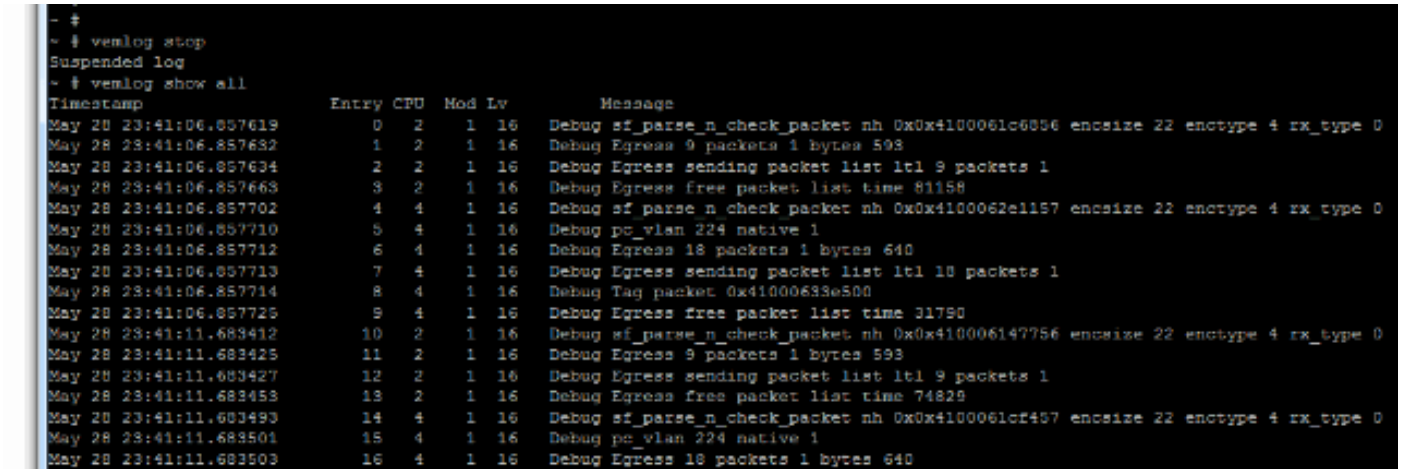
コマンドに[-]を追加すると、そのモジュールのデバッグレベルが削除されます。次のスクリーンショットに例を示します。



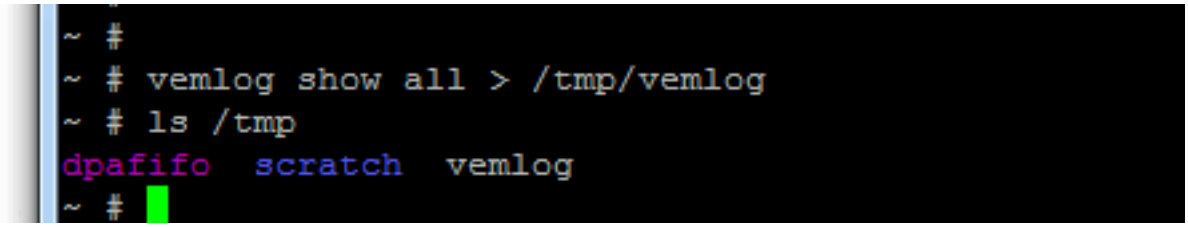
使用可能なすべてのデバッグモジュールは、**vemlog show debugコマンドで表示できます。**

監視するパッケージを指定したら、vemlog startコマンドを使用して**vemlogを起動**します。

ステップ 3：出力の表示



ログを表示する準備ができたら、**vemlog show allコマンドを発行**します。これにより、コンソール（またはSSHセッション）にすべてのログエントリが出力されます。 ホストに直接接続している場合は、これをファイルに送信して、後でコピーして分析することもできます。



このファイルを/vmfs/volumes/[*datastore_name*]ディレクトリに移動する場合は、ESXiのデータストアブラウザを使用してホストから移動できます。

vemlogもvmkernelログに保存されます。

ステップ 4：vemlogを通常のログレベルに復元する

```
~ # 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、vemlogを標準設定に復元します。これは、デバッグ情報が将来の問題に対して適切にキャプチャされるようにするために重要です。

vemlogはvmkernelログとvemlogのバッファの両方に保存されるため、特にチャティの高いデバッグは不要な情報で両方をフラッディングする可能性があります（特に有効のままにすると）。

関連情報

- テクニカル サポートとドキュメント – Cisco Systems