如何使用SNMP偵測和清除掛起的TCP連線

目錄

簡介

本文說明如何使用簡易網路管理通訊協定(SNMP)檢測和清除Cisco IOS裝置上的掛起TCP連線。本檔案也說明您用於此目的的SNMP物件。

名為PERL Script to Detect and Clear Hung TCP Connections一節提供了到實現這些指令的PERL指令碼的連結。

必要條件

需求

本文檔的讀者應瞭解以下主題：

- 瞭解如何檢視Cisco裝置上的TCP連線資訊
- SNMP **walk**、**get**、**get-next**和**set**命令的一般用法
- 瞭解如何在Cisco裝置上配置SNMP

採用元件

本檔案適用於執行IOS軟體(支援TCP-MIB和CISCO-TCP-MIB)的Cisco路由器和交換器。

註：NET-SNMP中預設不載入CISCO-TCP-MIB模組。如果未在系統上載入MIB模組，則必須使用OID引用對象而不是其名稱。

本檔案中的資訊是根據所有IOS軟體和硬體版本。

此資訊是根據以下版本的NET-SNMP而來：

- NET-SNMP版本5.1.2，網址為http://www.net-snmp.org/
已使用PERL版本測試了PERL指令碼：

- FreeBSD上的5.005_03
- Solaris 5.8上的5.8.0
- 5.005_02 — 作為Microsoft Windows 2000上的CiscoWorks SNMS的一部分提供
- Microsoft Windows 2000上的ActivePerl 5.8.4，可從 http://www.activestate.com/Products/ActivePerl/ 獲得。
本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路正在作用，請確保您已瞭解任何指令可能造成的影響。

慣例

如需文件慣例的詳細資訊，請參閱思科技術提示慣例。

背景資訊

MIB對象的詳細資訊 — 包括對象識別符號(OID)

以下是您使用的對象：

在CISCO-TCP-MIB模組中：

- ciscoTcpConnInBytes, OID .1.3.6.1.4.1.9.9.6.1.1.1.1此連線上輸入的位元組數。
- ciscoTcpConnInPkts, OID 1.3.6.1.4.1.9.9.6.1.1.1.2在此連線上輸入的資料包數。
- ciscoTcpConnOutBytes, OID .1.3.6.1.4.1.9.9.6.1.1.1.3此連線上輸出的位元組數
- ciscoTcpConnOutPkts, OID .1.3.6.1.4.1.9.9.6.1.1.1.4此連線上輸出的資料包數。
- ciscoTcpConnRetransPkts, OID .1.3.6.1.4.1.9.9.6.1.1.1.7在此連線上重新傳輸的資料包數。
- ciscoTcpConnRto, OID .1.3.6.1.4.1.9.9.6.1.1.1.9此連線的重新傳輸超時值。
在TCP-MIB模組中：

- tcpConnState, OID .1.3.6.1.2.1.6.13.1.1此連線的狀態。
Detailed MIB Object Information中提供了有關這些對象的更多詳細資訊。

使用SNMP檢測TCP連線是否掛起

摘要

以下步驟可幫助您確定TCP連線是否掛起：

1. 若要確定裝置中是否支援ciscoTcpConnRetransPkts和ciscoTcpConnRto對象，請對ciscoTcpConnRto執行SNMP **get-next**操作，並驗證是否返回了任何對象。**注意：**您只需要檢查一個對象，因為同時增加了這兩個對象的支援。**注意：**並非所有Cisco裝置都支援最後兩個對象(ciscoTcpConnRetransPkts**和**ciscoTcpConnRto)，**但是使用這些對象可以提高檢測的準確性。**如果支援ciscoTcpConnRetransPkts和ciscoTcpConnRto對象，請繼續執行步驟2。如果不支援ciscoTcpConnRetransPkts和ciscoTcpConnRto對象，請**繼續執行步驟3。**
2. 支援所有對象。對於每個TCP連線，請檢查以下專案：ciscoTcpConnOutBytes為0。ciscoTcpConnOutPkts為0。ciscoTcpConnRetransPkts大於0。ciscoTcpConnRto大於20,000。**附註：** 可以減少20,000以加快檢測。掛斷連線後，Rto大約需要一分鐘才能達到20,000。然而，較小的值可能會降低結果的精度。如果上述所有條件都成立，則此TCP連線將會掛起並且可以清除。繼續操作使用SNMP清除掛起的TCP連線。
3. 僅支援前四個對象。對於每個TCP連線，請檢查以下專案：ciscoTcpConnInBytes大於0。ciscoTcpConnInPkts為0。ciscoTcpConnOutBytes為0。ciscoTcpConnOutPkts為0。等待幾秒鐘，然**後再**次獲取對象，以驗證在建立過程中它是否不是TCP連線。**注意：前**兩個檢查（正數輸入位元組，但沒有輸入資料包）可能看起來很奇怪，但是根據許多裝置和IOS版本進行了驗證。**注意：支**持所有六個對象的IOS版本可能不會出現此行為，因此，步驟2中的測試不包括前兩個測試。如果所有對象兩次都滿足測試，則此TCP連線將掛起並且可以清除。繼續操作使用SNMP清除掛起的TCP連線。

逐步說明

此示例中的值為：

- 裝置主機名a = nms-7206a（支援所有對象）
- 裝置主機名b = nms-1605（僅支援前四個對象）
- 讀取社群=公共
- 寫入社群=專用

在這些命令中替換社群字串和主機名：

1. 確定此裝置是否支援ciscoTcpConnRetransPkts和ciscoTcpConnRto對象：對ciscoTcpConnRto執行**SNMP get-next**操作：
   ```
   snmpgetnext -c public nms-7206a ciscoTcpConnRto
   ```

   如果對象**受**支援，您會看到如下響應：
   ```
   CISCO-TCP-MIB::ciscoTcpConnRto.14.32.100.75.2065.172.18.86.111.23092 =
        INTEGER: 303 milliseconds
   ```

   註：用於這些對象的索引(本例中為14.32.100.75.2065.172.18.86.111.23092)是本地IP地址(14.32.100.75)、本地TCP埠號(2065)、遠端IP地址(172.18.86.111)和遠端TCP埠編號(TCP)的串聯23092。返回用於ciscoTcpConnRto。請繼續執行步驟2。如果對象**不**支援，您會看到如下響應：
   ```
   snmpgetnext -c public nms-1605 ciscoTcpConnRto
   CISCO-FLASH-MIB::ciscoFlashDevicesSupported.0 = INTEGER: 1
   ```

   返回不**用於**ciscoTcpConnRto對象。返回的精確對象並不重要。請繼續執行步驟3。
2. **獲取**有關支援Cisco TCP連線表中所有六個對象的裝置的每個TCP連線的資訊。對ciscoTcpConnOutBytes、ciscoTcpConnOutPkts、ciscoTcpConnRetransPkts和ciscoTcpConnRto執行SNMP get-next操作：

```
snmpgetnext -c public nms-7206a ciscoTcpConnOutBytes
                       ciscoTcpConnOutPkts
                       ciscoTcpConnRetransPkts
                       ciscoTcpConnRto
```

您會看到這樣的響應：

```
CISCO-TCP-MIB::ciscoTcpConnOutBytes.14.32.100.75.2065.172.18.86.111.23092 = Counter32:
383556
CISCO-TCP-MIB::ciscoTcpConnOutPkts.14.32.100.75.2065.172.18.86.111.23092 = Counter32: 8061
CISCO-TCP-MIB::ciscoTcpConnRetransPkts.14.32.100.75.2065.172.18.86.111.23092 = Counter32: 2
CISCO-TCP-MIB::ciscoTcpConnRto.14.32.100.75.2065.172.18.86.111.23092 = INTEGER: 303
milliseconds
```

驗證以下專案：ciscoTcpConnOutBytes為0。ciscoTcpConnOutPkts為0。ciscoTcpConnRetransPkts大於0。ciscoTcpConnRto大於20,000。**附註：** 可以減少20,000以加快檢測。掛斷連線後，Rto大約需要一分鐘才能達到20,000。然而，較小的值可能會降低結果的精度。如果以上所有情況均為真，則此TCP連線將掛起並且可以清除。繼續操作使用SNMP清除掛起的TCP連線。繼續**走**TCP連線表。為此，請在檢查掛起的連線時，使用返回的對象(例如：

```
snmpgetnext -c public nms-7206a ciscoTcpConnOutBytes.14.32.100.75.2065.172.18.86.111.23092
                       ciscoTcpConnOutPkts.14.32.100.75.2065.172.18.86.111.23092
                       ciscoTcpConnRetransPkts.14.32.100.75.2065.172.18.86.111.23092
                       ciscoTcpConnRto.14.32.100.75.2065.172.18.86.111.23092
```

使用上一個測試檢查每個條目，直到**get-next**操作按以下方式返回對象：

```
CISCO-TCP-MIB::ciscoTcpConnInPkts.14.32.100.75.2065.172.18.86.111.23092 = Counter32: 8097
CISCO-TCP-MIB::ciscoTcpConnElapsed.14.32.100.75.2065.172.18.86.111.23092 =
  Timeticks: (17296508) 2 days, 0:02:45.08
CISCO-TCP-MIB::ciscoTcpConnFastRetransPkts.14.32.100.75.2065.172.18.86.111.23092 =
Counter32: 0
CISCO-FLASH-MIB::ciscoFlashDevicesSupported.0 = INTEGER: 5
```

現在，您已遍歷此裝置上的所有TCP連線，並且您已經完成。

3. **獲**取有關僅支援Cisco TCP連線表中前四個對象的裝置的每個TCP連線的資訊。對ciscoTcpConnInBytes、ciscoTcpConnInPkts ciscoTcpConnOutBytes和ciscoTcpConnOutPkts執行SNMP **get-next**操作：

```
snmpgetnext -c public nms-1605 ciscoTcpConnInBytes
                       ciscoTcpConnInPkts
                       ciscoTcpConnOutBytes
                       ciscoTcpConnOutPkts
```

您會看到這樣的響應：

```
CISCO-TCP-MIB::ciscoTcpConnInBytes.14.32.6.185.23.14.32.100.33.2249 = Counter32: 68
CISCO-TCP-MIB::ciscoTcpConnInPkts.14.32.6.185.23.14.32.100.33.2249 = Counter32: 12
CISCO-TCP-MIB::ciscoTcpConnOutBytes.14.32.6.185.23.14.32.100.33.2249 = Counter32: 170
CISCO-TCP-MIB::ciscoTcpConnOutPkts.14.32.6.185.23.14.32.100.33.2249 = Counter32: 17
```

檢查這些是否正確：ciscoTcpConnInBytes大於0。ciscoTcpConnInPkts為0。ciscoTcpConnOutBytes為0。ciscoTcpConnOutPkts為0。請等待幾秒鐘，然**後再**次獲取對象。驗證在建立過程中它是否不是TCP連線。如果以上所有條件**均成立**，則此TCP連線將會掛起並且可以清除。繼續操作使用SNMP清除掛起的TCP連線。繼續**走**TCP連線表。為此，請在檢查掛起的連線時，使用返回的對象(例如：

```
snmpgetnext -c public nms-1605 ciscoTcpConnInBytes.14.32.6.185.23.14.32.100.33.2249
                       ciscoTcpConnInPkts.14.32.6.185.23.14.32.100.33.2249
                       ciscoTcpConnOutBytes.14.32.6.185.23.14.32.100.33.2249
```

ciscoTcpConnOutPkts.14.32.6.185.23.14.32.100.33.2249

使用上一個測試檢查每個條目，直到**get-next**操作按以下方式返回對象：

```
CISCO-TCP-MIB::ciscoTcpConnOutBytes.14.32.6.185.23.14.32.100.33.4184 = Counter32: 170
CISCO-TCP-MIB::ciscoTcpConnOutPkts.14.32.6.185.23.14.32.100.33.4184 = Counter32: 17
CISCO-TCP-MIB::ciscoTcpConnInPkts.14.32.6.185.23.14.32.100.33.4184 = Counter32: 12
CISCO-TCP-MIB::ciscoTcpConnElapsed.14.32.6.185.23.14.32.100.33.4184 = Timeticks: (4345)
0:00:43.45
```

現在，您已遍歷此裝置上的所有TCP連線，並且您已經完成。

使用SNMP清除掛起的TCP連線

逐步說明

您可以使用SNMP清除掛起的TCP連線。SNMP命令等效於clear tcp local <*local_ip*> <*local_port*> ***remote*** <*remote_ip*> <*remote_port*>命令。用於清除線路的對象為**tcpConnState**。

若要使用SNMP清除掛起的TCP連線，請發出以下命令：

```
snmpset -c private nms-7206a tcpConnState.14.32.100.75.2065.172.18.86.111.23092 integer
deleteTCB
TCP-MIB::tcpConnState.14.32.100.75.2065.172.18.86.111.23092 = INTEGER: deleteTCB(12)
```

註：用於這些對象的索引(本例中為14.32.100.75.2065.172.18.86.111.23092)是本地IP地址(14.32.100.75)、本地TCP埠號(2065)、遠端IP地址(172.18.86.111)和遠端TCP埠編號(TCP)的串聯23092。

附註： 您必須使用在使用SNMP檢測TCP連線是否掛起中確定的確切索引。請注意，此命令會斷開TCP連線，而不會發出警告。

詳細的MIB對象資訊

```
.1.3.6.1.4.1.9.9.6.1.1.1.1
ciscoTcpConnInBytes OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Counter
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "Number of bytes that have been input on this TCP
                        connection."
::= { ciscoTcpConnEntry 1 }

.1.3.6.1.4.1.9.9.6.1.1.1.2
ciscoTcpConnOutBytes OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Counter
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "Number of bytes that have been output on this TCP
                        connection."
::= { ciscoTcpConnEntry 2 }

.1.3.6.1.4.1.9.9.6.1.1.1.3
```

```
ciscoTcpConnInPkts OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Counter
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "Number of packets that have been input on this TCP
                        connection."
::= { ciscoTcpConnEntry 3 }

.1.3.6.1.4.1.9.9.6.1.1.1.4
ciscoTcpConnOutPkts OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Counter
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "Number of packets that have been output on this TCP
                        connection."
::= { ciscoTcpConnEntry 4 }

.1.3.6.1.4.1.9.9.6.1.1.1.7
ciscoTcpConnRetransPkts OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Counter
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "The total number of packets retransmitted due to a timeout -
                        that is, the number of TCP segments transmitted containing
                        one or more previously transmitted octets."
::= { ciscoTcpConnEntry 7 }

.1.3.6.1.4.1.9.9.6.1.1.1.9
ciscoTcpConnRto OBJECT-TYPE
        -- FROM CISCO-TCP-MIB
        SYNTAX          Integer
        MAX-ACCESS      read-only
        STATUS          Current
        DESCRIPTION     "The current value used by a TCP implementation for the
                        retransmission timeout."
::= { ciscoTcpConnEntry 9 }

.1.3.6.1.2.1.6.13.1.1
tcpConnState OBJECT-TYPE
        -- FROM RFC1213-MIB
        SYNTAX          Integer { closed(1), listen(2), synSent(3), synReceived(4),
                        established(5), finWait1(6), finWait2(7), closeWait(8), lastAck(9),
                        closing(10), timeWait(11), deleteTCB(12) }
        MAX-ACCESS      read-write
        STATUS          Mandatory
        DESCRIPTION     "The state of this TCP connection.

                        The only value which may be set by a management
                        station is deleteTCB(12).  Accordingly, it is
                        appropriate for an agent to return a `badValue'
                        response if a management station attempts to set
                        this object to any other value.

                        If a management station sets this object to the
                        value deleteTCB(12), then this has the effect of
                        deleting the TCB (as defined in RFC 793) of the
                        corresponding connection on the managed node,
                        resulting in immediate termination of the
                        connection.

                        As an implementation-specific option, a RST
```

```
                        segment may be sent from the managed node to the
                        other TCP endpoint (note however that RST segments
                        are not sent reliably)."
::= { tcpConnEntry 1 }
```

用於檢測和清除掛起TCP連線的PERL指令碼

此連結提供包含PERL指令碼和必要MIB模組的歸檔檔案。按一下右鍵該連結並將檔案儲存到系統。

- fixTCPhang.tgz

檔案中的檔案包括：

- bin/fixTCPhang.pl
- mibs/CISCO-SMI.my
- mibs/CISCO-TCP-MIB.my

要提取指令碼和MIB模組，請在類UNIX作業系統上使用gzip和tar等實用程式。例如，要將檔案解壓到**/tmp**,**假設**歸檔檔案放在/tmp中，**請執行以下操作**:

```
cd /tmp; gzip -dc fixTCPhang.tgz | tar -xvf -
```

注意：您可能需要編輯指令碼的第一行以指定PERL的位置。

在Microsoft Windows作業系統上使用winzip或其他實用程式解壓檔案。如果將檔案解壓到**c:\tmp**，則運行指令碼時不必指定 — m選項。

使用以下命令呼叫檔案：

```
fixTCPhang.pl -c public -C private -f nms-7206a
```

對於找到的每個掛起TCP連線，您都會看到類似以下輸出的行：

```
Found bad TCP connection: Local IP: 14.32.100.75 port 23 Remote IP: 172.18.100.33 port 47878:
CLEARED
```

當提供了讀寫社群字串並指定了 — f選項時，指令碼將清除連線。請注意輸出末尾的CLEARED語句。

指令碼支援SNMP版本1、2c和3。如果指定SNMP版本3，則必須在 — v引數中指定所有身份驗證資訊。以下是使用SNMP v3的範例：

```
fixTCPhang.pl -v "3 -a MD5 -u chelliot -A chelliot -l authNoPriv" -f nms-dmz-ap1200-b
```

用於為上一個示例配置SNMP v3的IOS命令如下：

```
snmp-server group chelliot-group v3 auth write v1default
snmp-server user chelliot chelliot-group v3 auth md5 chelliot
```

注意：在此測試中使用的Windows版本的NET-SNMP中似乎存在錯誤。錯誤不允許SHA驗證正常運作。

還有幾個其他選項可用於此指令碼。一些指令碼選項包括查詢NET-SNMP命令列實用程式的位置，以及MIB模組(如果它們不在/tmp/mib)的查詢**位置。**您還可以檢視這些選項的摘要：

```
fixTCPhang.pl
fixTCPhang.pl [-dfhV -c <read_community> -C <write_community> -m <mib_directory>
               -p <command_path> -t <timeout> -v <snmp_version>] <device>
  Version 1.2
  Detect hung TCP connections on <device>, optionally clearing them.
  Options:  -c  Specify read community string. Defaults to public.
            -C  Specify the readwrite community string. No default.
                 Must be supplied for the script to clear hung connections.
            -d  Turn on debug mode.
            -f  Fix or clear any hung TCP connections found.
            -h  Print this message.
            -m  Specify the directory to find CISCO-SMI.my and CISCO-TCP-MIB.my.
                 Defaults to /tmp/mibs.
            -p  Where to find the net-snmp utilities.
                 Optional if the utilities are in the path.
            -t  SNMP Timeout value. Defaults to 5 sec.
            -v  Specify SNMP version to use: One of 1, 2c, or 3.
                 If 3 is specified then this option must include all of the
                 authentication information for SNMPv3. For example:
                 "3 -a MD5 -u chelliot -A chelliot -l authNoPriv"
                 Note: NET-SNMP seems to have a bug with SHA authentication on Windows.
                 See the NET-SNMP documentation for more information.
                 Defaults to SNMP version 1.
            -V  Print version number.
```

相關資訊

- 技術支援 - Cisco Systems