如何使用SNMP检测和清除挂起的TCP连接

目录

简介

本文档介绍如何使用简单网络管理协议(SNMP)检测和清除Cisco IOS设备上挂起的TCP连接。本文档还说明了您用于此目的的SNMP对象。

标题为“PERL Script to Detect and Clear Hung TCP Connections”的部分提供了指向实现这些说明的PERL脚本的链接。

先决条件

要求

本文档的读者应掌握以下这些主题的相关知识：

- 了解如何查看思科设备上的TCP连接信息
- SNMP walk、**get**、**get-next**和**set**命令**的一般用途**
- 了解如何在思科设备上配置SNMP

使用的组件

本文档适用于运行IOS软件的Cisco路由器和交换机，支持TCP-MIB和CISCO-TCP-MIB模块。

注意：NET-SNMP中默认不加载CISCO-TCP-MIB模块。如果系统上未加载MIB模块，则必须使用OID引用对象而非其名称。

本文档中的信息基于所有IOS软件和硬件版本。

该信息基于此版本的NET-SNMP:

- 在 http://www.net-snmp.org/ 可上获取 NET-SNMP 版本 5.1.2

使用PERL版本测试了PERL脚本：

- 5.005_03（在FreeBSD上）
- Solaris 5.8上的5.8.0
- 5.005_02 — 作为Microsoft Windows 2000上CiscoWorks SNMS的一部分发货
- Microsoft Windows 2000上的ActivePerl 5.8.4，可从 http://www.activestate.com/Products/ActivePerl/获取。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

规则

有关文件规则的更多信息请参见“ Cisco技术提示规则”。

背景信息

MIB对象的详细信息 — 包括对象标识符(OID)

以下是您使用的对象：

从CISCO-TCP-MIB模块：

- ciscoTcpConnInBytes, OID .1.3.6.1.4.1.9.9.6.1.1.1此连接上输入的字节数。
- ciscoTcpConnInPkts,OID 1.3.6.1.4.1.9.9.6.1.1.2此连接上输入的数据包数。
- ciscoTcpConnOutBytes, OID .1.3.6.1.4.1.9.9.6.1.1.3此连接上的字节数输出
- ciscoTcpConnOutPkts,OID .1.3.6.1.4.1.9.9.6.1.1.4此连接上输出的数据包数。
- ciscoTcpConnRetransPkts, OID .1.3.6.1.4.1.9.9.6.1.1.7此连接上重新传输的数据包数。
- ciscoTcpConnRto、OID .1.3.6.1.4.1.9.9.6.1.1.9此连接的重新传输超时值。

从TCP-MIB模块：

- tcpConnState, OID .1.3.6.1.2.1.6.13.1.1此连接的状态。

详细MIB对象信息中提供了有关这些对象的详细信息。

使用SNMP检测TCP连接是否挂起

摘要

以下步骤可帮助您确定TCP连接是否挂起：

1. 要确定设备中是否支持ciscoTcpConnRetransPkts和ciscoTcpConnRto对象，请对ciscoTcpConnRto执行SNMP **get-next**操作，并验证是否返回任何对象。**注意：**您只需检查一个对象，因为同时添加了对这两个对象的支持。**注意：**并非所有Cisco设备都支持最后两个对象(ciscoTcpConnRetransPkts和ciscoTcpConnRto)，但使用它们可以提高检测的准确性。如果ciscoTcpConnRetransPkts和ciscoTcpConnRto对象受支持，请继续执行步骤2。如果ciscoTcpConnRetransPkts和ciscoTcpConnRto对象不受**支持**，请继续执行步骤3。
2. 支持所有对象。对于每个TCP连接，请检查以下内容：ciscoTcpConnOutBytes为0。ciscoTcpConnOutPkts为0。ciscoTcpConnRetransPkts大于0。ciscoTcpConnRto大于20,000。**注意：** 可以减少20,000，以加快检测速度。连接中断后， Rto大约需要一分钟才能达到20,000。但是，较小的值可能会降低结果的准确性。如果以前的全部都为真，则此TCP连接将挂起并可以清除。继续使用SNMP清除挂起的TCP连接。
3. 仅支持前四个对象。对于每个TCP连接，请检查以下内容：ciscoTcpConnInBytes大于0。ciscoTcpConnInPkts为0。ciscoTcpConnOutBytes为0。ciscoTcpConnOutPkts为0。等待几秒，**再**次获取对象，以验证它在建立过程中不是TCP连接。**注意：**前两项检查（输入字节数为正数，但没有输入数据包）似乎很奇怪，但它们已针对大量设备和IOS版本进行验证。**注意：**支持所有六个对象的IOS版本可能不显示此行为，因此，步骤2中的测试不包括前两项测试。如果所有对象都同时满足测试，则此TCP连接将挂起并可以清除。继续使用SNMP清除挂起的TCP连接。

逐步指导

本示例中的值为：

- 设备主机名a = nms-7206a（支持所有对象）
- 设备主机名b = nms-1605（仅支持前四个对象）
- Read community = public
- Write community = private

在以下命令中替换社区字符串和主机名：

1. 确定此设备是否支持ciscoTcpConnRetransPkts和ciscoTcpConnRto对象：对**ciscoTcpConnRto执行SNMP** get-next操作:
   ```
   snmpgetnext -c public nms-7206a ciscoTcpConnRto
   ```

   如果支持**对**象，您会看到如下响应：
   ```
   CISCO-TCP-MIB::ciscoTcpConnRto.14.32.100.75.2065.172.18.86.111.23092 =
        INTEGER: 303 milliseconds
   ```

   注意：在本例中，用于这些对象的索引`14.32.100.75.2065.172.18.86.111.23092`是本地IP地址 — 14.32.10的串联0.75、本地TCP端口号 — 2065、远程IP地址 — 172.18.86.111和远程TCP端口号 — 23092。返回的是ciscoTcpConnRto。继续执行步骤 2。如果对象不**支**持，您会看到如下响应：
   ```
   snmpgetnext -c public nms-1605 ciscoTcpConnRto
   CISCO-FLASH-MIB::ciscoFlashDevicesSupported.0 = INTEGER: 1
   ```

   返回的**不是**ciscoTcpConnRto对象。返回的确切对象并不重要。继续执行步骤 3。
2. **获取**支持Cisco TCP连接表中所有六个对象的设备的每个TCP连接的信息。对ciscoTcpConnOutBytes、ciscoTcpConnOutPkts、ciscoTcpConnRetransPkts和ciscoTcpConnRoto执行SNMP get-next操作：

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

您会看到这样的响应：

```
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
```

验证以下内容：ciscoTcpConnOutBytes为0。ciscoTcpConnOutPkts为0。ciscoTcpConnRetransPkts大于0。ciscoTcpConnRto大于20,000。**注意：** 可以减少20,000，以加快检测速度。连接中断后， Rto大约需要一分钟才能达到20,000。但是，较小的值可能会降低结果的准确性。如果所有这些都为真，则此TCP连接将挂起并可以清除。继续使用SNMP清除挂起的TCP连接。继续**浏览**TCP连接表。为此，在检查挂起连接时，请使**用返回的**对象（如以下对象）重复执行SNMP get-next操作：

```
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
```

使用上一测试检查每个条目，直到**get-next操**作以此方式返回对象：

```
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
```

现在，您已浏览此设备上的所有TCP连接，并且已完成。

3. **获取**有关仅支持Cisco TCP连接表中前四个对象的设备的每个TCP连接的信息。对ciscoTcpConnInBytes、ciscoTcpConnInPkts ciscoTcpConnOutBytes和ciscoTcpConnOutPkts执行SNMP **get-next操作：**

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

您会看到这样的响应：

```
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
```

检查以下信息是否正确：ciscoTcpConnInBytes大于0。ciscoTcpConnInPkts为0。ciscoTcpConnOutBytes为0。ciscoTcpConnOutPkts为0。等几秒钟，再**次**获取对象。验证它是否不是正在建立的TCP连接。如果上述所有**均**为true，则此TCP连接将挂起并可以清除。继续使用SNMP清除挂起的TCP连接。继续**浏览**TCP连接表。为此，在检查挂起连接时，请使**用返回的**对象（如以下对象）重复执行SNMP get-next操作：

```
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

使用上一测试检查每个条目，直到**get-next操**作以此方式返回对象：

```
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
```

现在，您已浏览此设备上的所有TCP连接，并且已完成。

使用SNMP清除挂起的TCP连接

逐步指导

您可以使用SNMP清除挂起的TCP连接。SNMP命令等同于clear tcp local <*local_ip*> <*local_port*> remote <*remote_ip*> <*remote_port*>*命令*。用于清除线路的对象是**tcpConnState**。

要清除与SNMP的挂起TCP连接，请发出以下命令：

```
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)
```

注意：在本例中，用于这些对象的索引`14.32.100.75.2065.172.18.86.111.23092`是本地IP地址 — 14.32.10的串联0.75、本地TCP端口号 — 2065、远程IP地址 — 172.18.86.111和远程TCP端口号 — 23092。

注意： 必须使用您确定在使用SNMP检测TCP连接是否挂起时挂起的确切索引。请注意，此命令会断开TCP连接，但不会发出警告。

详细的MIB对象信息

```
.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 }
```

检测并清除挂起的TCP连接的PERL脚本

此链接提供包含PERL脚本和必要MIB模块的存档文件。右键点击链接，将文件保存到系统。

- fixTCPhang.tgz

存档中的文件包括：

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

要解压脚本和MIB模块，请在类似UNIX的操作系统上使用实用程序，如gzip和tar。例如，要将文件解压到/tmp，假设存档文件放在/tmp中：

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

注意：您可能需要编辑脚本的第一行以指定PERL的位置。

在Microsoft Windows操作系统上使用winzip或其他实用程序解压文件。如果将文件解压到c:\tmp，则运行脚本时无需指定 — m选项。

使用以下命令调用文件：

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

对于找到的每个挂起的TCP连接，您会看到类似以下输出的一行：

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

由于提供了读写社区字符串并指定了 — f选项，脚本清除了连接。请注意CLEARED语句在输出末尾。

脚本支持SNMP版本1、2c和3。如果指定SNMP版本3，则必须在 — v参数中指定所有身份验证信息。以下是使用SNMP v3的示例：

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

为上例配置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：此测试中使用的NET-SNMP的Windows版本中似乎存在Bug。Bug不允许SHA身份验证正常工作。

您还可以使用此脚本使用其他几个选项。一些脚本选项包括在何处查找NET-SNMP命令行实用程序以及在何处查找MIB模块(如果它们不在/tmp/mibs**中)**。您还可以查看这些选项的以下摘要：

```
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.
```

相关信息

- 技术支持 - Cisco Systems