

Erkennen und Löschen von Hung TCP-Verbindungen mithilfe von SNMP

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Konventionen](#)

[Hintergrundinformationen](#)

[Details der MIB-Objekte - Enthält OIDs \(Object Identifiers\)](#)

[Verwenden von SNMP zum Erkennen einer TCP-Verbindung](#)

[Zusammenfassung](#)

[Schrittweise Anleitung](#)

[Verwenden von SNMP zum Löschen einer sich ändernden TCP-Verbindung](#)

[Schrittweise Anleitung](#)

[Detaillierte MIB-Objektinformationen](#)

[PERL-Skript zum Erkennen und Löschen von Hung TCP-Verbindungen](#)

[Zugehörige Informationen](#)

[Einführung](#)

In diesem Dokument wird beschrieben, wie das Simple Network Management Protocol (SNMP) zum Erkennen und Löschen von TCP-Verbindungen auf einem Cisco IOS-Gerät verwendet wird. Im Dokument werden auch die zu diesem Zweck verwendeten SNMP-Objekte erläutert.

Der Abschnitt mit dem Titel [PERL Script to Detect and Clear Hung TCP Connections \(PERL-Skript zum Erkennen und Löschen von Hung TCP-Verbindungen\)](#) enthält einen Link zu einem PERL-Skript, das diese Anweisungen implementiert.

[Voraussetzungen](#)

[Anforderungen](#)

Die Leser dieses Dokuments sollten folgende Themen kennen:

- Informationen zum Anzeigen von TCP-Verbindungsinformationen auf Cisco Geräten
- Allgemeine Verwendung von SNMP-Befehlen für **Walking**, **Get**, **Get-Next** und **Set**-Befehle
- Sie wissen, wie SNMP auf einem Cisco Gerät konfiguriert wird.

Verwendete Komponenten

Dieses Dokument gilt für Cisco Router und Switches mit IOS-Software, die die [TCP-MIB](#)- und die [CISCO-TCP-MIB](#)-Module unterstützt.

Hinweis: Das CISCO-TCP-MIB-Modul wird standardmäßig nicht im NET-SNMP geladen. Wenn das MIB-Modul nicht auf Ihr System geladen ist, müssen Sie die OID verwenden, um auf ein Objekt anstatt auf dessen Namen zu verweisen.

Die Informationen in diesem Dokument basieren auf allen IOS-Software- und -Hardwareversionen.

Die Informationen basieren auf dieser Version von NET-SNMP:

- NET-SNMP Version 5.1.2 verfügbar unter <http://www.net-snmp.org/> 

Das PERL-Skript wurde mit PERL-Versionen getestet:

- 5.005_03 auf FreeBSD
- 5.8.0 unter Solaris 5.8
- 5.005_02 - im Rahmen von CiscoWorks SNMS unter Microsoft Windows 2000 ausgeliefert
- ActivePerl 5.8.4 unter Microsoft Windows 2000, verfügbar unter <http://www.activestate.com/Products/ActivePerl/> 

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie in den [Cisco Technical Tips Conventions](#).

Hintergrundinformationen

Details der MIB-Objekte - Enthält OIDs (Object Identifiers)

Dies sind die Objekte, die Sie verwenden:

Vom [CISCO-TCP-MIB](#)-Modul:

- [ciscoTcpConnInBytes](#), OID .1.3.6.1.4.1.9.9.6.1.1.1.1Die Anzahl der Bytes, die in dieser Verbindung eingegeben werden.
- [ciscoTcpConnInPkts](#), OID 1.3.6.1.4.1.9.9.6.1.1.1.2Die Anzahl der über diese Verbindung eingegebenen Pakete.
- [ciscoTcpConnOutBytes](#), OID .1.3.6.1.4.1.9.9.6.1.1.1.3Die Anzahl der Byte, die in dieser Verbindung ausgegeben werden
- [ciscoTcpConnOutPkts](#), OID .1.3.6.1.4.1.9.9.6.1.1.1.4Die Anzahl der Pakete, die für diese Verbindung ausgegeben werden.
- [ciscoTcpConnRetransPkts](#), OID .1.3.6.1.4.1.9.9.6.1.1.1.7Die Anzahl der über diese Verbindung erneut übertragenen Pakete

- [ciscoTcpConnRto](#), OID .1.3.6.1.4.1.9.9.6.1.1.1.9 Der Timeoutwert für die erneute Übertragung dieser Verbindung.

Vom [TCP-MIB](#)-Modul:

- [tcpConnState](#), OID .1.3.6.1.2.1.6.13.1.1 Der Status dieser Verbindung.

Weitere Details zu diesen Objekten finden Sie in den [detaillierten MIB-Objektinformationen](#).

Verwenden von SNMP zum Erkennen einer TCP-Verbindung

Zusammenfassung

Mithilfe dieser Schritte können Sie feststellen, ob eine TCP-Verbindung hängt:

1. Um festzustellen, ob die [Objekte ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#) im Gerät unterstützt werden, führen Sie einen SNMP **get-next** Vorgang auf [ciscoTcpConnRto](#) durch, und überprüfen Sie, ob Objekte zurückgegeben werden. **Hinweis:** Sie müssen nur ein Objekt überprüfen, da Unterstützung für beide gleichzeitig hinzugefügt wurde. **Hinweis:** Nicht alle Cisco Geräte unterstützen die letzten beiden Objekte ([ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#)), ihre Verwendung kann jedoch die Genauigkeit der Erkennung erhöhen. Wenn die [Objekte ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#) unterstützt werden, fahren Sie mit Schritt 2 fort. Wenn die Objekte [ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#) nicht unterstützt werden, fahren Sie mit Schritt 3 fort.
2. Alle Objekte werden unterstützt. Überprüfen Sie für jede TCP-Verbindung Folgendes: [ciscoTcpConnOutBytes](#) ist 0. [ciscoTcpConnOutPkts](#) ist 0. [ciscoTcpConnRetransPkts](#) ist größer als 0. [ciscoTcpConnRto](#) ist größer als 20.000. **Hinweis:** Die 20.000 können reduziert werden, um die Erkennung zu beschleunigen. Es dauert etwa eine Minute, bis Rto 20.000 erreicht, sobald die Verbindung nicht mehr reagiert. Kleinere Werte können jedoch die Genauigkeit des Ergebnisses verringern. Wenn alle vorherigen Bedingungen zutreffen, wird diese TCP-Verbindung nicht mehr reagiert und kann gelöscht werden. Fahren Sie mit [SNMP zum Löschen einer sich ändernden TCP-Verbindung fort](#).
3. Nur die ersten vier Objekte werden unterstützt. Überprüfen Sie für jede TCP-Verbindung Folgendes: [ciscoTcpConnInBytes](#) ist größer als 0. [ciscoTcpConnInPkts](#) ist 0. [ciscoTcpConnOutBytes](#) ist 0. [ciscoTcpConnOutPkts](#) ist 0. Warten Sie einige Sekunden, und **rufen Sie** die Objekte erneut auf, um zu überprüfen, ob es sich bei der Herstellung nicht um eine TCP-Verbindung handelt. **Hinweis:** Die ersten beiden Überprüfungen (eine positive Anzahl von Eingabebytes, aber keine Eingabepakete) mögen seltsam erscheinen, wurden jedoch für zahlreiche Geräte und IOS-Versionen verifiziert. **Hinweis:** IOS-Versionen, die alle sechs Objekte unterstützen, zeigen dieses Verhalten möglicherweise nicht an. Daher sind diese ersten beiden Tests im Test in Schritt 2 nicht enthalten. Wenn alle Objekte die Tests beide Mal erfüllen, wird diese TCP-Verbindung nicht mehr reagiert und kann gelöscht werden. Fahren Sie mit [SNMP zum Löschen einer sich ändernden TCP-Verbindung fort](#).

Schrittweise Anleitung

Die Werte in diesem Beispiel sind:

- Geräte-Hostname a = nms-7206a (unterstützt alle Objekte)

- Geräte-Hostname b = nms-1605 (unterstützt nur die ersten vier Objekte)
- Community lesen = öffentlich
- Community schreiben = privat

Ersetzen Sie die Community-Strings und den Hostnamen in den folgenden Befehlen:

1. Stellen Sie fest, ob dieses Gerät die Objekte [ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#) unterstützt:Führen Sie einen **SNMP Get-Next**-Vorgang auf [ciscoTcpConnRto](#) aus:

```
snmpgetnext -c public nms-7206a ciscoTcpConnRto
```

Wenn die Objekte unterstützt **werden**, wird eine Antwort wie folgt angezeigt:

```
CISCO-TCP-MIB::ciscoTcpConnRto.14.32.100.75.2065.172.18.86.111.23092 =
    INTEGER: 303 milliseconds
```

Hinweis: Der für diese Objekte verwendete Index, in diesem Fall

14.32.100.75.2065.172.18.86.111.23092, ist eine Verkettung der lokalen IP-Adresse 14.32.10.75, die lokale TCP-Portnummer 2065, die Remote-IP-Adresse 172.18.86.111 und die Remote-TCP-Portnummer 23092.Die Rücksendung gilt für [ciscoTcpConnRto](#). Fahren Sie mit Schritt 2 fort.Wenn die Objekte **nicht** unterstützt werden, sehen Sie eine Antwort wie diese:

```
snmpgetnext -c public nms-1605 ciscoTcpConnRto
CISCO-FLASH-MIB::ciscoFlashDevicesSupported.0 = INTEGER: 1
```

Die Rückgabe gilt **nicht** für das [ciscoTcpConnRto](#)-Objekt. Das genaue zurückgegebene Objekt ist nicht wichtig. Fahren Sie mit Schritt 3 fort.

2. **Hier erhalten Sie** Informationen zu jeder TCP-Verbindung für Geräte, die alle sechs Objekte in der Cisco TCP-Verbindungstabelle unterstützen.Durchführen eines **SNMP Get-Next**-Vorgangs auf [ciscoTcpConnOutBytes](#), [ciscoTcpConnOutPkts](#), [ciscoTcpConnRetransPkts](#) und [ciscoTcpConnRto](#):

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

Sie sehen eine Antwort wie diese:

```
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
```

Überprüfen Sie Folgendes:[ciscoTcpConnOutBytes](#) ist 0.[ciscoTcpConnOutPkts](#) ist 0.[ciscoTcpConnRetransPkts](#) ist größer als 0.[ciscoTcpConnRto](#) ist größer als 20.000.**Hinweis:** Die 20.000 können reduziert werden, um die Erkennung zu beschleunigen. Es dauert etwa eine Minute, bis Rto 20.000 erreicht, sobald die Verbindung nicht mehr reagiert. Kleinere Werte können jedoch die Genauigkeit des Ergebnisses verringern.Wenn alle diese Bedingungen zutreffen, wird diese TCP-Verbindung nicht mehr reagiert und kann gelöscht werden. Fahren Sie mit [SNMP zum Löschen einer sich ändernden TCP-Verbindung fort](#).Gehen Sie weiter zur TCP-Verbindungstabelle. Führen Sie hierzu wiederholt einen **SNMP Get-Next**-Vorgang durch, während Sie nach hängenden Verbindungen suchen. Verwenden Sie dabei die folgenden Objekte:

```
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
```

Überprüfen Sie jeden Eintrag mithilfe des vorherigen Tests, bis der **get-next**-Vorgang Objekte auf diese Weise zurückgibt:

```
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
```

Sie haben jetzt alle TCP-Verbindungen auf diesem Gerät durchlaufen und sind fertig.

3. **Hier erhalten Sie** Informationen zu jeder TCP-Verbindung für Geräte, die nur die ersten vier Objekte in der Cisco TCP-Verbindungstabelle unterstützen. Durchführen eines SNMP **get-next**-Vorgangs auf [ciscoTcpConnInBytes](#), [ciscoTcpConnInPkts](#), [ciscoTcpConnOutBytes](#) und [ciscoTcpConnOutPkts](#):

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

Sie sehen eine Antwort wie diese:

```
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
```

Überprüfen Sie, ob diese zutreffend sind: [ciscoTcpConnInBytes](#) ist größer als 0. [ciscoTcpConnInPkts](#) ist 0. [ciscoTcpConnOutBytes](#) ist 0. [ciscoTcpConnOutPkts](#) ist 0. Warten Sie einige Sekunden, und **holen Sie** die Objekte erneut. Stellen Sie sicher, dass es sich nicht um eine TCP-Verbindung handelt, die gerade hergestellt wird. Wenn alle oben genannten Bedingungen zutreffen, wird diese TCP-Verbindung nicht mehr reagiert und kann gelöscht werden. Fahren Sie mit [SNMP zum Löschen einer sich ändernden TCP-Verbindung fort](#). Gehen Sie weiter zur TCP-Verbindungstabelle. Führen Sie hierzu wiederholt einen SNMP **Get-Next**-Vorgang durch, während Sie nach hängenden Verbindungen suchen. Verwenden Sie dabei die folgenden Objekte:

```
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
```

Überprüfen Sie jeden Eintrag mithilfe des vorherigen Tests, bis der **get-next**-Vorgang Objekte auf diese Weise zurückgibt:

```
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
```

Sie haben jetzt alle TCP-Verbindungen auf diesem Gerät durchlaufen und sind fertig.

Verwenden von SNMP zum Löschen einer sich ändernden TCP-Verbindung

Schrittweise Anleitung

Sie können SNMP verwenden, um eine unterbrochene TCP-Verbindung zu löschen. Der SNMP-Befehl entspricht dem Befehl `clear tcp local <local_ip> <local_port> remote <remote_ip> <remote_port>`. Das Objekt, das Sie zum Löschen einer Zeile verwenden, ist `tcpConnState`.

Führen Sie folgenden Befehl aus, um eine unterbrochene TCP-Verbindung mit SNMP zu löschen:

```
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)
```

Hinweis: Der für diese Objekte verwendete Index, in diesem Fall

14.32.100.75.2065.172.18.86.111.23092, ist eine Verkettung der lokalen IP-Adresse 14.32.10.75, die lokale TCP-Portnummer 2065, die Remote-IP-Adresse 172.18.86.111 und die Remote-TCP-Portnummer 23092.

Hinweis: Sie müssen den exakten Index verwenden, der unter [SNMP zum Erkennen einer TCP-Verbindung verwenden](#) als **fehlerhaft** ermittelt wurde. Beachten Sie, dass dieser Befehl eine TCP-Verbindung ohne Vorwarnung trennt.

Detaillierte MIB-Objektinformationen

```
.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 }
```

[PERL-Skript zum Erkennen und Löschen von Hung TCP-](#)

Verbindungen

Dieser Link stellt eine Archivdatei mit einem PERL-Skript und den erforderlichen MIB-Modulen bereit. Klicken Sie mit der rechten Maustaste auf den Link, und speichern Sie die Datei auf Ihrem System.

- [fixTCPPhang.tgz](#)

Die Dateien im Archiv sind:

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

Um das Skript und die MIB-Module zu extrahieren, verwenden Sie ein Dienstprogramm wie gzip und tar auf UNIX-ähnlichen Betriebssystemen. So extrahieren Sie beispielsweise die Dateien nach **/tmp** unter der Annahme, dass die Archivdatei in **/tmp** abgelegt wird:

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

Hinweis: Möglicherweise müssen Sie die erste Zeile des Skripts bearbeiten, um den Speicherort von PERL anzugeben.

Verwenden Sie winzip oder andere Dienstprogramme auf Microsoft Windows-Betriebssystemen, um die Dateien zu extrahieren. Wenn Sie die Dateien nach **c:\tmp** extrahieren, müssen Sie beim Ausführen des Skripts nicht die Option **-m** angeben.

Rufen Sie die Dateien mit dem folgenden Befehl auf:

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

Für jede unterbrochene TCP-Verbindung wird eine Zeile wie die folgende Ausgabe angezeigt:

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

Während der Read-Write Community String angegeben und die **-f**-Option angegeben wurde, wurde die Verbindung vom Skript gelöscht. Beachten Sie die **CLEARED**-Anweisung am Ende der Ausgabe.

Das Skript unterstützt die SNMP-Versionen 1, 2c und 3. Wenn Sie SNMP Version 3 angeben, müssen Sie alle Authentifizierungsinformationen im **-v**-Argument angeben. Dies ist ein Beispiel für die Verwendung von SNMP v3:

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

Die IOS-Befehle zum Konfigurieren von SNMP v3 für das vorherige Beispiel sind:

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

Hinweis: Es scheint ein Fehler in der in diesem Test verwendeten Windows-Version von NET-SNMP zu geben. Der Fehler erlaubt nicht, dass die SHA-Authentifizierung ordnungsgemäß funktioniert.

Es gibt mehrere andere Optionen, die Sie mit diesem Skript verwenden können. Zu den Skriptoptionen gehören, wo die NET-SNMP-Befehlszeilendienstprogramme gesucht werden sollen und wo die MIB-Module zu finden sind, wenn sie sich nicht in **/tmp/mibs** befinden. Sie können auch diese Zusammenfassung der folgenden Optionen anzeigen:

fixTCPPhang.pl

```
fixTCPPhang.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.

[Zugehörige Informationen](#)

- [Technischer Support - Cisco Systems](#)