Ping MIB Implementation

Document ID: 13383

Contents

Introduction
Prerequisites
   Requirements
   Components Used
Conventions
Example Script
   The Script
Related Information

Introduction

The Ping Group is part of the Cisco proprietary MIB under the Cisco Management branch (.1.3.6.1.4.1.9.9.16.). The Ping Group can be used to set up, perform, and retrieve Internet Control Message Protocol (ICMP) activity between remote devices from a management station.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

Example Script

You can use the example script to initiate this sequence on HP OpenView or NetView. You can also enter these commands as **snmpsets** and **snmpgets** using the GUIs of the network management platforms. Use one of the following methods to access the GUI:

- From the menu bar, select **Tools**, then **MIB Browser: SNMP**.
- At the command line type **xmbrowser**.
The Script

Management_Station---------Router_Source---------Router_Dest

```
<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
</table>
| `echo "###### Create the instance ######"

```

```

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.16.333 integer 6</code></td>
</tr>
</tbody>
</table>

```

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
</table>
| `echo "###### Now let's set the characteristics of the ping ######"

```

<table>
<thead>
<tr>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.16.333 integer 5</code></td>
</tr>
</tbody>
</table>

```
Only the first three sets below are REQUIRED. The rest have default values.

Set ciscoPingEntryOwner = any_name

snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.15.333 octetstring any_name

Set ciscoPingProtocol = 1 = ip (see CISCO-TC-V1SMI.my CiscoNetworkProtocol)

snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.2.333 integer 1

Set ciscoPingAddress = #.#.#.##--take Remote_Dest's ip & convert each octet to hex

snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.3.333 octetstringhex AB 44 76 67

Set the packet count to 20 (ciscoPingPacketCount)

snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.4.333 integer 20

Set the packetsize to 100 (ciscoPingPacketSize)

snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.5.333 integer 100

echo

echo "##### Now let's verify that the ping is ready to go and launch it #######"

Get ciscoPingEntryStatus and make sure it is now equal to 2. This means

# notInService which indicates that we're ready to go.

snmpget -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.16.333
# Set ciscoPingEntryStatus = 1 to tell it to activate.

```bash
snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.16.333 integer 1
```

```bash
echo
```

```bash
echo "##### Let's look at the results. #####"
```

```bash
snmpwalk -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1
```

```bash
echo
```

```bash
echo "##### Now that we've gotten the results, let's destroy the row #####"
```

```bash
snmpset -c public Router_Source .1.3.6.1.4.1.9.9.16.1.1.1.16.333 integer 6
```

## Related Information

- Technical Support – Cisco Systems