Troubleshoot SNMP Basic Functionality

Contents

Introduction

Prerequisites

- Requirements
- Components Used

Troubleshoot Most Common Errors

Related Information

Introduction

This document describes some Simple Network Management Protocol (SNMP) troubleshooting scenarios.

Prerequisites

Requirements

Cisco recommends that you have knowledge of:

- SNMPv2c and SNMPv3 Protocol Operation
- Cisco IOS® XE Command Line Interface (CLI)

Components Used

The information in this document is based on these software and hardware versions:

- Cisco Catalyst 3650 Series Switch

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, ensure that you understand the potential impact of any command.

Troubleshoot Most Common Errors

1. Error Message: "%SNMP-3-RESPONSE_DELAYED: processing GetNext of <OID> (<elapsed-time> msecs)"

In this example, ciscoMgmt.810.1.2.1.1 is the OID associated with the delayed GetNext request. Correlate repeated messages with CPU utilization, NMS polling records, and release-specific defects before identifying the root cause:

```
*May 24 01:30:48.463: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.2.1.1 (24008 msec

*May 24 01:31:12.477: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.2.1.1 (24012 msec

*May 24 01:31:36.486: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.2.1.1 (24008 msec

*May 24 01:32:00.503: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.3.1.1 (24016 msec

*May 24 01:32:24.515: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.3.1.1 (24012 msec

*May 24 01:32:48.528: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.3.1.1 (24012 msec

*May 24 01:33:12.537: %SNMP-3-RESPONSE_DELAYED: processing GetNext of ciscoMgmt.810.1.3.1.1 (24008 msec
```

To troubleshoot:

Check SNMP configuration on the device. For SNMPv2c, it looks similar to the output if multiple communities are added to device:

```
snmp-server community TAC1 RO

snmp-server community TAC2 RO  -->
```

For SNMPv3:

```
snmp-server view TESTV3 iso include

snmp-server group TestGroupV3 v3 auth read TESTV3

snmp-server user cisco TestGroupV3 v3 auth md5 ciscorules priv des56 cisco123
```

Enter the configuration mode of the device and add a view to the SNMP configuration.

For SNMPv2c:

```
snmp-server community TAC1 RO view cutdown RO

snmp-server community TAC2 RO view cutdown RO
```

The Idea is to exclude the OID causing the issue, however, please review what is the function of the OID that can be excluded before applying any changes:

```
snmp-server view cutdown iso included

snmp-server view cutdown ciscoMgmt.810 excluded -->>>
```

For SNMPv3 the view exclusion is applied with the **snmp-server group** command:

```
snmp-server view TESTV3 internet included

snmp-server view TESTV3 ciscoMgmt.810 excluded

snmp-server group TestGroupV3 v3 priv write TESTV3
```

2. Error Message "High CPU Utilization due to SNMP Flash Cache".

<#root>

```
Device#

show processes cpu sorted
```

```
CPU utilization for five seconds: 99%/0%; one minute: 22%; five minutes: 18%

PID    Runtime(ms)     Invoked      uSecs      5Sec          1Min     5Min     TTY    Process

447    561399          143012       3925       0.00%         1.58%    1.83%     0     Snmp Flash Cache
```

SNMP Logs:

%SYS-2-SIGPENDING: Multiple Signals are sent to a process 91 -Process= "Snmp Flash Cache", ipl= 0, pid= 91.

```
888888888888888888888888888888888888888888888898878889

625424254283314655456532533533772205363424335694492379

  100                                              *      *

  90 * *    *  *    *** ***  *  *  **  * *      ***    **

  80 ************************************************************

  70 ************************************************************

  60 ************************************************************

  50 ************************************************************

  40 ############################################################

  30 ############################################################

  20 ############################################################

  10 ############################################################

   0....5....1....1....2....2....3....3....4....4....5....5....6....6....7..
```

To workaround:

The SNMP Flash Cache process can consume high CPU when Flash MIB caching is enabled. Confirm the configured state with **show running-config all | include snmp mib flash cache** and validate process CPU before applying **no snmp mib flash cache**.

3. Error Message: "%SNMP-3-INPUT_QFULL_ERR:Packet dropped due to input queue full"

A possible reason for a queue full error can be heavy polling on the device or a specific OID that causes the issue. To mitigate this, first, check if the device is heavily polled.

In order to do so, execute this command:

<#root>

```
Device#
```

show snmp stats oid

```
time-stamp                              #of times requested           OID
15:40:19 BKK Dec 27 2019        11180008                           ifAlias
15:40:19 BKK Dec 27 2019        44018183                          dot1dBasePortEntry.4
15:40:19 BKK Dec 27 2019        44018212                          dot1dBasePortEntry.3
15:40:19 BKK Dec 27 2019        45216156                          ipNetToPhysicalEntry.4
15:40:19 BKK Dec 27 2019        44018059                          dot1dBasePortEntry.5
15:40:19 BKK Dec 27 2019        44578303                          dot1dBasePortEntry.1
15:40:19 BKK Dec 27 2019        6011756                          dot3StatsEntry.19
15:40:19 BKK Dec 27 2019        11095925                          ifSpeed
15:40:19 BKK Dec 27 2019        12879927                          dot1dTpFdbEntry.3
15:40:19 BKK Dec 27 2019        84535                              vmMembershipSummaryEntry.2
15:40:19 BKK Dec 27 2019        3241107                            vmMembershipSummaryEntry.3
15:40:19 BKK Dec 27 2019        45208908                          ipNetToMediaEntry.2
15:40:19 BKK Dec 27 2019        45223410                          ipNetToPhysicalEntry.6
15:40:19 BKK Dec 27 2019        44018324                          dot1dBasePortEntry.2
```

To troubleshoot:

You need to change the settings on your NMS and reduce the polling intervals for the device. Once the polling interval is reduced, queue full error must be mitigated. If not, then you need to check for the OID that causes the issue. To find the OID that causes the issue and to troubleshoot on the same, please refer to the previously mentioned Error Message 1.

4. Error Message: "High CPU utilization due to SNMP ENGINE".

Identify the issue:

The router suffers a high CPU at the time it is polled by a client, and this can be checked with the **show process cpu <sorted>** command at the time of the high CPU. You can see that the SNMP Engine process

takes all the CPU resources:

```
<#root>

Device#

show processes cpu sorted


CPU utilization for five seconds: 99%/0%; one minute: 22%; five minutes: 18%

PID    Runtime(ms)      Invoked      uSecs   5Sec          1Min              5Min    TTY    Process

189     1535478456      697105815    2202    88.15%        13.40%            8.74%    0     SNMP ENGINE
```

The problematic OID causes the high CPU to be slower than the others, which could also cause some timeout when the client requests this OID. Most of the methods attempt to find the OID that provides a slower answer. That is because they are the most likely to cause the high CPU. Once the OID is identified, you can lock that respective OID in order to mitigate the errors.

Note: If none of the methods listed here help to identify an OID that causes the issue, please open a case with TAC.

Method 1. Use **show snmp stats oid** command.

The **show snmp stats oid** command displays the last OID that was polled. It displays the timestamp in order, the goal is to identify the OID that did respond slowly. This command is also helpful if you want to find what MIBs are polled more often by the client.

```
<#root>

Device#

show snmp stats oid


time-stamp #of times     requested   OI

14:34:38 CET Oct 25 2020 24       atEntry.2

14:34:29 CET Oct 25 2020 40       atEntry.1

14:34:11 CET Oct 25 2020  11       ifOutErrors
```

```
14:34:07 CET Oct 25 2020 10       ifOutDiscards

14:34:06 CET Oct 25 2020 10       ifOutUcastPkts

14:34:06 CET Oct 25 2020 10      ifOutOctets

14:34:05 CET Oct 25 2020 10      ifInUnknownProtos
```

You can see that Entry.1 took 18 seconds to get computed. This suggests that the CPU was busy in order to compute this data.

Method 2. Observe the SNMP client.

In order to find the OID that is responsible for the high CPU usage on the device, you can initiate a snmpwalk to a device from an NMS server and observe the output. The OIDs that respond slower than the other OIDs can be the ones responsible for high CPU utilization.

To troubleshoot:

Check SNMP configuration on the device. For SNMPv2, it needs to look like:

```
snmp-server community TAC1 RO

snmp-server community TAC2 RO
```

- For SNMPv3

```
snmp-server view TESTV3 iso include

snmp-server group TestGroupV3  v3 auth read TESTV3

snmp-server user <username> TestGroupV3 v3 auth sha <auth-secret> priv aes 128 <priv-secret>
```

Note: Use unique secrets that comply with the organizational credential policy. Confirm SHA and AES syntax and support for the documented Cisco IOS XE release.

Enter the configuration mode of the device and add a view to the SNMP configuration to change it.

- For SNMPv2

```
snmp-server community  TAC1 RO view cutdown RO

snmp-server community TAC2 RO view cutdown RO
```

Add these lines in the configuration mode. The Idea is to exclude the OID that causes the issue, however, please read out what is the function of the OID that you are about to exclude:

```
snmp-server view cutdown iso included

snmp-server view cutdown <oid-subtree> excluded
```

Related Information

- Configuration Guide: Chapter: SNMP
- Cisco Technical Support & Downloads