Using SNMP to Find a Port Number from a MAC Address on a Catalyst Switch

Document ID: 44800

Introduction

This document describes how to use Simple Network Management Protocol (SNMP) to obtain the port number on a Cisco Catalyst switch from which you know the MAC address.

Prerequisites

Requirements

Readers of this document should have knowledge of these topics:

- How to get VLANs from a Catalyst switch with use of SNMP
- How to use community string indexing with SNMP
- General use of the SNMP **get** command and **walk** command

Components Used

This document applies to Catalyst switches that run regular Catalyst OS (CatOS) or Cisco IOS® Software. The software supports the BRIDGE−MIB and the IF−MIB.

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

- Catalyst 3524XL that runs Cisco IOS Software Release 12.0(5)WC5a
- Net−SNMP version 5.0.6

Note: To obtain this software, refer to Net−SNMP .

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.

Background

For more information on how to query the content−addressable memory (CAM) table, VLANs, and all related MIBs, such as the CISCO−VTP−MIB and the BRIDGE−MIB, refer to the *Background* section of the document How To Get Dynamic CAM Entries (CAM Table) for Catalyst Switches Using SNMP.

Details of the MIB Variables, Which Includes Object Identifiers (OIDs)

```
.1.3.6.1.2.1.17.4.3.1.1
dot1dTpFdbAddress OBJECT−TYPE
         −− FROM BRIDGE−MIB
         −− TEXTUAL CONVENTION MacAddress
         SYNTAX          OCTET STRING (6)
         MAX−ACCESS      read−only
         STATUS          Mandatory
         DESCRIPTION    "A unicast MAC address for which the bridge has forwarding
                 and/or filtering information."
::= { iso(1) org(3) dod(6) internet(1) mgmt(2) mib−2(1) dot1dBridge(17) dot1dTp(4)
dot1dTpFdbTable(3) dot1dTpFdbEntry(1) 1 }

.1.3.6.1.2.1.17.4.3.1.2
dot1dTpFdbPort OBJECT−TYPE
         −− FROM BRIDGE−MIB
         SYNTAX          Integer
         MAX−ACCESS      read−only
         STATUS          Mandatory
         DESCRIPTION    "Either the value "0", or the port number of the port on which
                 a frame having a source
                 address equal to the value of the corresponding instance of
                 dot1dTpFdbAddress has been seen.
                 A value of "0" indicates that the port number has not been learned,
                 but that the bridge does
                 have some forwarding/filtering information about this address (that is,
                 in the StaticTable).
                       Implementors are encouraged to assign the port value to this
                 object whenever it is
                 learned, even for addresses for which the corresponding value of
                 dot1dTpFdbStatus is not learned(3)."
::= { iso(1) org(3) dod(6) internet(1) mgmt(2) mib−2(1) dot1dBridge(17) dot1dTp(4)
dot1dTpFdbTable(3) dot1dTpFdbEntry(1) 2 }

.1.3.6.1.2.1.2.2.1.1
ifIndex OBJECT−TYPE
    SYNTAX      InterfaceIndex
    MAX−ACCESS  read−only
    STATUS      current
    DESCRIPTION      "A unique value, greater than zero, for each interface.  It
            is recommended that values are assigned contiguously
            starting from 1.  The value for each interface sub−layer
            must remain constant at least from one re−initialization of
            the entity's network management system to the next re−
            initialization."
    ::= { ifEntry 1 }

.1.3.6.1.2.1.17.1.4.1.2
dot1dBasePortIfIndex OBJECT−TYPE
              SYNTAX  INTEGER
              ACCESS  read−only
```

```
                STATUS  mandatory
                DESCRIPTION
                        "The value of the instance of the ifIndex object,
                        defined in MIB-II, for the interface corresponding
                        to this port."
                ::= { dot1dBasePortEntry 2 }

.1.3.6.1.2.1.31.1.1.1.1
ifName OBJECT-TYPE
    SYNTAX      DisplayString
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION    "The textual name of the interface.  The value of this
            object should be the name of the interface as assigned by
            the local device and should be suitable for use in commands
            entered at the device's `console'.  This might be a text
            name, such as `le0' or a simple port number, such as `1',
            depending on the interface naming syntax of the device.  If
            several entries in the ifTable together represent a single
            interface as named by the device, then each will have the
            same value of ifName.  Note that for an agent which responds
            to SNMP queries concerning an interface on some other
            (proxied) device, then the value of ifName for such an
            interface is the proxied device's local name for it.
            If there is no local name, or this object is otherwise not
            applicable, then this object contains a zero-length string."
    ::= { ifXEntry 1 }
```

Get the Port Number on Which a MAC Address Has Been Learned

Step-by-Step Instructions

Complete the steps in this section in order to use SNMP to get the port number on which a MAC address has been learned. Consider that the port number is in VLAN1.

Note: In the commands in this section:

- **`public`** is the read community string.
- **`@1`** is the VLAN 1 part of the read community string.
- **`crumpy`** is the device host name.

  Note: You can also use the IP address for this host name.

Note: The Conclusion section uses the values that appear in ***italics*** in the command output.

1. Retrieve the VLANs. Use the **snmpwalk** command on the vtpVlanState object (.1.3.6.1.4.1.9.9.46.1.3.1.1.2):

   ```
   %snmpwalk -c public crumpy .1.3.6.1.4.1.9.9.46.1.3.1.1.2
   CISCO-VTP-MIB::vtpVlanState.1.1 = INTEGER: operational(1)
   CISCO-VTP-MIB::vtpVlanState.1.3 = INTEGER: operational(1)
   CISCO-VTP-MIB::vtpVlanState.1.7 = INTEGER: operational(1)
   CISCO-VTP-MIB::vtpVlanState.1.10 = INTEGER: operational(1)
   ...
   ```

   Note: This command uses community string indexing. The command also uses vtpVlanState, which has OID `.1.3.6.1.4.1.9.9.46.1.3.1.1.2`. If you have loaded the MIBs to your network management system (NMS), you can use the object name instead of the OID. Issue this command

instead:

```
%snmpwalk -c public@1 crumpy vtpVlanState
```

Note: You can also use the object names in steps 2 through 6.

2. Issue this command in order to obtain the MAC address table by considering that the port belongs to VLAN1:

```
snmpwalk -c public@1 crumpy .1.3.6.1.2.1.17.4.3.1.1

17.4.3.1.1.0.0.12.7.172.8 =  Hex: 00 00 0C 07 AC 08
17.4.3.1.1.0.1.2.27.80.145 =  Hex: 00 01 02 1B 50 91
17.4.3.1.1.0.1.3.72.77.90 =  Hex: 00 01 03 48 4D 5A
17.4.3.1.1.0.1.3.72.221.191 =  Hex: 00 01 03 48 DD BF
...
```

Note: Provide the appropriate VLAN number after the community string. In this example, it is VLAN1.

The command lists all MAC addresses that have been learned on all ports that belong to VLAN 1.

3. Issue this command to determine the bridge port number for VLAN 1:

```
snmpwalk -c public@1 crumpy .1.3.6.1.2.1.17.4.3.1.2

17.4.3.1.2.0.0.12.7.172.8 = 13
17.4.3.1.2.0.1.2.27.80.128 = 13
17.4.3.1.2.0.1.2.27.80.145 = 13
17.4.3.1.2.0.1.2.163.145.225 = 13
...
```

Note: VLAN 1 is dot1dTpFdbPort , or `.1.3.6.1.2.1.17.4.3.1.2`.

4. Issue this command to map the bridge port to the ifIndex, OID `.1.3.6.1.2.1.2.2.1.1`:

```
snmpwalk -c public@1 crumpy .1.3.6.1.2.1.17.1.4.1.2

17.1.4.1.2.13 = 2
17.1.4.1.2.14 = 3
17.1.4.1.2.15 = 4
17.1.4.1.2.16 = 5
```

This command queries the dot1dBasePortIfIndex, which has OID `.1.3.6.1.2.1.17.1.4.1.2`.

5. Use the **walk** command with ifName in order to correlate the ifIndex value with a correct port name.

Issue this command:

Note: The ifName has OID `.1.3.6.1.2.1.31.1.1.1.1`.

```
snmpwalk -c public@1 crumpy .1.3.6.1.2.1.31.1.1.1.1

ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.1 = VL1
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.2 = Fa0/1
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.3 = Fa0/2
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.4 = Fa0/3
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.5 = Fa0/4
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.6 = Fa0/5
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.7 = Fa0/6
...
```

6. Link a MAC address to the port on which the address was learned.

- From Step 1, the MAC address is:

```
17.4.3.1.1.0.0.12.7.172.8 = Hex: 00 00 0C 07 AC 08
```

- From Step 2, the bridge port tells that the MAC address belongs to bridge port number 13:

```
17.4.3.1.2.0.0.12.7.172.8 = 13
```

- From Step 3, the bridge port number 13 has ifIndex number 2:

```
17.1.4.1.2.13 = 2
```

- From Step 4, the ifIndex 2 corresponds to port Fast Ethernet 0/1:

```
ifMIB.ifMIBObjects.ifXTable.ifXEntry.ifName.2 = Fa0/1
```

Conclusion

The MAC address 00 00 0C 07 AC 08 is learned on port Fa0/1.

Compare this conclusion with output from:

- The **show cam dynamic** command for CatOS switches
- The **show mac** command for Cisco IOS Software switches

Here is the sample output:

```
crumpy# show mac
Dynamic Address Count:                  58
Secure Address Count:                   2
Static Address (User-defined) Count:    0
System Self Address Count:              51
Total MAC addresses:                    111
Maximum MAC addresses:                  8192
Non-static Address Table:
Destination Address  Address Type  VLAN  Destination Port
-------------------  ------------  ----  -------------------

0000.0c07.ac08       Dynamic          1  FastEthernet0/1

0001.021b.5091       Dynamic          1  FastEthernet0/1
0001.0348.4d5a       Dynamic          1  FastEthernet0/1
0001.0348.ddbf       Dynamic          1  FastEthernet0/1
0001.972d.dfae       Dynamic          1  FastEthernet0/1
0002.55c6.cfe7       Dynamic          1  FastEthernet0/1
0002.7d61.d400       Dynamic          1  FastEthernet0/1
```

Related Information

- **SNMP Object Navigator**
- **Technical Support & Documentation − Cisco Systems**

Updated: Oct 26, 2005 Document ID: 44800