

SNMP を使った、Catalyst スイッチの MAC アドレスからのポート番号の検索

目次

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[背景説明](#)

[オブジェクト識別子 \(OID\) を含む、MIB 変数の詳細](#)

[MAC アドレスが判明しているポート番号の取得](#)

[手順説明](#)

[関連情報](#)

概要

このドキュメントでは、MAC アドレスがわかっている Cisco Catalyst スイッチのポート番号を取得するために簡易ネットワーク管理プロトコル (SNMP) を使用する方法について説明します。

前提条件

要件

このドキュメントの読者は次のトピックについて理解している必要があります。

- SNMP を使用して Catalyst スイッチから VLAN を取得する方法
- SNMP でのコミュニティ スtring インデックスを使用する方法
- SNMP の `get` コマンドと `walk` コマンドの一般的な使用法

使用するコンポーネント

このドキュメントは、標準の Catalyst OS (CatOS) または Cisco IOS® ソフトウェアを実行する Catalyst スイッチに適用されます。ソフトウェアは [BRIDGE-MIB](#) および [IF-MIB](#) をサポートします。

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- Cisco IOS ソフトウェア リリース 12.0(5)WC5a が稼働する Catalyst 3524XL
- Net-SNMP バージョン 5.0.6注: このソフトウェアを入手するには、[Net-SNMP](#) を参照してください。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

表記法

ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

背景説明

連想メモリ（CAM）テーブル、VLAN、および関連するすべての MIB（CISCO-VTP-MIB、BRIDGE-MIB など）のクエリー方法の詳細については、『[SNMP を使った、Catalyst スイッチのダイナミック CAM エントリ（CAM テーブル）の取得方法](#)』の「[背景説明](#)」の項を参照してください。

オブジェクト識別子（OID）を含む、MIB 変数の詳細

.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 }
```

MAC アドレスが判明しているポート番号の取得

手順説明

SNMP を使用して、MAC アドレスが判明しているポート番号を取得するには、この項の手順を実行します。ポート番号は VLAN1 内にあると考えます。

注: この項のコマンドでは、

- **public** は、読み取り用コミュニティ ストリングです。
- **@@1** は、読み取り用コミュニティ ストリングの VLAN 1 部分です。
- **crumpy** は、デバイスのホスト名です。注: このホスト名に IP アドレスも使用できます。

注: 「[結論](#)」の項では、コマンド出力の**斜体**で表示される値を使用します。

1. VLAN を取得します。vtpVlanState オブジェクトに対して **snmpwalk** コマンドを使用します (.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) ... 注: このコマンドは、コミュニティ ストリング インデックスを
```

使用します。また、OID が .1.3.6.1.4.1.9.9.46.1.3.1.1.2 である [vtpVlanState](#) も使用します。ネットワーク管理システム (NMS) に MIB をロードした場合は、OID ではなくオブジェクト名を使用できます。代わりに、次のコマンドを発行します。

```
%snmpwalk -c public@1 crumpy vtpVlanState 注: また、ステップ 2 から 6 では、オブジェクト名も使用できます。
```

2. ポートが VLAN1 に属すると考えた場合、MAC アドレス テーブルを取得するには、次のコマンドを発行します。

```
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 ... 注: コミュニティストリングの後に、適切な VLAN 番号を指定します。この例では、VLAN1 です。このコマンドにより、VLAN 1 に属するすべてのポートで判明したすべての MAC アドレスがリストされます。
```

3. VLAN 1 のブリッジ ポート番号を設定するために、次のコマンドを発行します。

```
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 ... 注: VLAN 1 は、dot1dTpFdbPort、すなわち、.1.3.6.1.2.1.17.4.3.1.2 です。
```

4. ブリッジ ポートを [ifIndex](#)、OID .1.3.6.1.2.1.2.2.1.1 にマップするために、次のコマンドを発行します。

```
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 このコマンドは、OID が .1.3.6.1.2.1.17.1.4.1.2 である dot1dBasePortIfIndex を照会します。
```

5. [ifIndex](#) 値を正しいポート名に関連付けるには、[ifName](#) を指定して walk コマンドを使用します。次のコマンドを発行します。注: [ifName](#) の OID は .1.3.6.1.2.1.31.1.1.1.1 です。

```
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. アドレスが判明したポートに MAC アドレスをリンクします。ステップ 1 から、MAC アドレスが次のように判明します。17.4.3.1.1.0.0.12.7.172.8 = Hex: 00 00 0C 07 AC 08 ステップ 2 から、ブリッジ ポートにより、MAC アドレスがブリッジ ポート番号 13 に属することがわかります。17.4.3.1.2.0.0.12.7.172.8 = 13 ステップ 3 から、ブリッジ ポート番号 13 の ifIndex 番号が 2 であることが判明します。17.1.4.1.2.13 = 2 ステップ 4 から、ifIndex 2 はファースト イーサネット 0/1 ポートに対応していることがわかります。

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

結論

MAC アドレス 00 00 0 C 07 AC 08 が、ポート Fa0/1 で判明しています。

この結論を、以下の出力と比較します。

- CatOS スイッチの `show cam dynamic` コマンド
- Cisco IOS ソフトウェア スイッチの `show mac` コマンド

次に出カ例を示します。

```
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 ...

[関連情報](#)

- [SNMP Object Navigator](#)
- [テクニカルサポートとドキュメント - Cisco Systems](#)