Cómo agregar, modificar y eliminar VLAN en un Catalyst con SNMP

Contenido

Introducción

Este documento describe cómo crear y eliminar las VLAN en un switch Cisco Catalyst que use Simple Network Management Protocol (SNMP). También describe cómo agregar puertos a una VLAN mediante SNMP.

Prerequisites

Requirements

Antes de utilizar la información de este documento, asegúrese de comprender:

- Cómo funcionan ifTable y ifIndexes
- Cómo funcionan las VLAN en los switches Cisco Catalyst
- Cómo ver la información de VLAN en los switches Cisco Catalyst
- El uso general de los comandos **get**, **set** y **walk** SNMP

Componentes

Este documento es para switches Catalyst que ejecutan Catalyst OS o Catalyst IOS regulares que soportan IF-MIB, CISCO-VTP-MIB y CISCO-VLAN-MEMBERSHIP-MIB. La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Catalyst 3524XL que ejecuta CatIOS 12.0(5)WC5a
- NET-SNMP version 5.0.6 available at http://www.net-snmp.org/

La información que se presenta en este documento se originó a partir de dispositivos dentro de un ambiente de laboratorio específico. All of the devices used in this document started with a cleared (default) configuration. Si está trabajando en una red activa, antes de utilizar cualquier comando asegúrese de comprender el impacto potencial de cualquier comando.

Convenciones

Para obtener más información sobre las convenciones del documento, consulte Convenciones de Consejos Técnicos de Cisco.

Background

Detalles de las variables MIB: incluidos los identificadores de objetos (OID)

```
1.3.6.1.4.1.9.9.46.1.3.1.1.2 (CISCO-VTP-MIB)
vtpVlanState  OBJECT-TYPE
    SYNTAX     INTEGER { operational(1),
                         suspended(2),
                         mtuTooBigForDevice(3),
                         mtuTooBigForTrunk(4) }
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION            "The state of this VLAN.

            The state 'mtuTooBigForDevice' indicates that this device
            cannot participate in this VLAN because the VLAN's MTU is
            larger than the device can support.

            The state 'mtuTooBigForTrunk' indicates that while this
            VLAN's MTU is supported by this device, it is too large for
            one or more of the device's trunk ports."
    ::= { vtpVlanEntry 2 }


1.3.6.1.4.1.9.9.46.1.4.1.1.1 (CISCO-VTP-MIB)
vtpVlanEditOperation OBJECT-TYPE
    SYNTAX     INTEGER { none(1),
                         copy(2),
                         apply(3),
                         release(4),
                         restartTimer(5)
                       }
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "This object always has the value 'none' when read.  When
            written, each value causes the appropriate action:

             'copy' - causes the creation of rows in the
            vtpVlanEditTable exactly corresponding to the current global
```

```
            VLAN information for this management domain.  If the Edit
            Buffer (for this management domain) is not currently empty,
            a copy operation fails.  A successful copy operation starts
            the deadman-timer.

             'apply' - first performs a consistent check on the the
            modified information contained in the Edit Buffer, and if
            consistent, then tries to instanciate the modified
            information as the new global VLAN information.  Note that
            an empty Edit Buffer (for the management domain) would
            always result in an inconsistency since the default VLANs
            are required to be present.

             'release' - flushes the Edit Buffer (for this management
            domain), clears the Owner information, and aborts the
            deadman-timer.  A release is generated automatically if the
            deadman-timer ever expires.

             'restartTimer' - restarts the deadman-timer.

             'none' - no operation is performed."
    ::= { vtpEditControlEntry 1 }


1.3.6.1.4.1.9.9.46.1.4.1.1.3 (CISCO-VTP-MIB)
vtpVlanEditBufferOwner OBJECT-TYPE
    SYNTAX     OwnerString
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "The management station which is currently using the Edit
            Buffer for this management domain.  When the Edit Buffer for
            a management domain is not currently in use, the value of
            this object is the zero-length string.  Note that it is also
            the zero-length string if a manager fails to set this object
            when invoking a copy operation."
    ::= { vtpEditControlEntry 3 }


1.3.6.1.4.1.9.9.46.1.4.2.1.11 (CISCO-VTP-MIB)
vtpVlanEditRowStatus OBJECT-TYPE
    SYNTAX     RowStatus
1:active
2:notInService
3:notReady
4:createAndGo
5:createAndWait
6:destroy
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "The status of this row.  Any and all columnar objects in an
            existing row can be modified irrespective of the status of
            the row.

            A row is not qualified for activation until instances of at
            least its vtpVlanEditType, vtpVlanEditName and
            vtpVlanEditDot10Said columns have appropriate values.

            The management station should endeavor to make all rows
            consistent in the table before 'apply'ing the buffer.  An
            inconsistent entry in the table will cause the entire
            buffer to be rejected with the vtpVlanApplyStatus object
            set to the appropriate error value."
    ::= { vtpVlanEditEntry 11 }
```

```
1.3.6.1.4.1.9.9.46.1.4.2.1.3.1.48 (CISCO-VTP-MIB)
vtpVlanEditType OBJECT-TYPE
    SYNTAX     VlanType
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "The type which this VLAN would have.
            An implementation may restrict access to this object."
    DEFVAL     { ethernet }
    ::= { vtpVlanEditEntry 3 }


1.3.6.1.4.1.9.9.46.1.4.2.1.4.1.48 (CISCO-VTP-MIB)
vtpVlanEditName OBJECT-TYPE
    SYNTAX     DisplayString (SIZE (1..32))
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "The name which this VLAN would have.  This name would be
            used as the ELAN-name for an ATM LAN-Emulation segment of
            this VLAN.

            An implementation may restrict access to this object."
    ::= { vtpVlanEditEntry 4 }


1.3.6.1.4.1.9.9.46.1.4.2.1.6.1.48 (CISCO-VTP-MIB)
vtpVlanEditDot10Said OBJECT-TYPE
    SYNTAX     OCTET STRING (SIZE (4))
    MAX-ACCESS read-create
    STATUS     current
    DESCRIPTION            "The value of the 802.10 SAID field which would be used for
            this VLAN.

            An implementation may restrict access to this object."
    ::= { vtpVlanEditEntry 6 }


1.3.6.1.4.1.9.9.46.1.4.1.1.2.1 (CISCO-VTP-MIB)
vtpVlanApplyStatus OBJECT-TYPE
    SYNTAX     INTEGER { inProgress(1),
                         succeeded(2),
                         configNumberError(3),
                         inconsistentEdit(4),
                         tooBig(5),
                         localNVStoreFail(6),
                         remoteNVStoreFail(7),
                         editBufferEmpty(8),
                         someOtherError(9)
                       }
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION            "The current status of an 'apply' operation to instanciate
            the Edit Buffer as the new global VLAN information (for this
            management domain).  If no apply is currently active, the
            status represented is that of the most recently completed
            apply.  The possible values are:

               inProgress - 'apply' operation in progress;

               succeeded - the 'apply' was successful (this value is
                      also used when no apply has been invoked since the
                      last time the local system restarted);

               configNumberError - the apply failed because the value of
```

```
                    vtpVlanEditConfigRevNumber was less or equal to
                    the value of current value of
                    managementDomainConfigRevNumber;

                inconsistentEdit - the apply failed because the modified
                    information was not self-consistent;

                tooBig - the apply failed because the modified
                    information was too large to fit in this VTP
                    Server's non-volatile storage location;

                localNVStoreFail - the apply failed in trying to store
                    the new information in a local non-volatile
                    storage location;

                remoteNVStoreFail - the apply failed in trying to store
                    the new information in a remote non-volatile
                    storage location;

                editBufferEmpty - the apply failed because the Edit
                    Buffer was empty (for this management domain).

                someOtherError - the apply failed for some other reason
                    (e.g., insufficient memory)."
    ::= { vtpEditControlEntry 2 }


1.3.6.1.4.1.9.9.68.1.2.2.1.2 (CISCO-VLAN-MEMBERSHIP-MIB)
vmVlan  OBJECT-TYPE
        SYNTAX     INTEGER(0..4095)
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION                "The VLAN id of the VLAN the port is assigned to
                when vmVlanType is set to static or dynamic.
                This object is not instantiated if not applicable.

                The value may be 0 if the port is not assigned
                to a VLAN.

                If vmVlanType is static, the port is always
                assigned to a VLAN and the object may not be
                set to 0.

                If vmVlanType is dynamic the object's value is
                0 if the port is currently not assigned to a VLAN.
                In addition, the object may be set to 0 only."
        ::= { vmMembershipEntry 2 }
```

Agregar una VLAN a un switch Cisco Catalyst con SNMP

Step-by-Step Instructions

En el ejemplo que se muestra a continuación, la VLAN 11 se agrega al switch:

1. Para verificar qué VLAN están configuradas actualmente en el switch, emita una **snmpwalk** en el OID **vtpVlanState**:**Nota:** El último número del OID es el número de VLAN.

   ```
   snmpwalk -c public crumpy vtpVlanState
   cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1
   ```

```
.1 : INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1
.48 : INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1
.1002 : INTEGER: operational
```

2. Verifique si la edición está siendo utilizada por otra estación o dispositivo NMS. La edición no está en uso si ve este mensaje: `no hay objetos MIB contenidos en subárbol`:

```
snmpwalk -c public crumpy vtpVlanEditTable
no MIB objects contained under subtree.
```

3. La edición no está en uso, por lo que es seguro empezar a editarla. Establezca el **vtpVlanEditOperation** en el estado de copia (número entero 2). Esto le permite crear la VLAN.

```
snmpset -c private crumpy vtpVlanEditOperation.1 integer 2
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpEditControlTable.vtpEditControlEntry.
vtpVlanEditOperation.1 : INTEGER: copy
```

4. Para hacer visible el propietario actual del permiso de edición, puede establecer el propietario cuando ejecute el comando, **vtpVlanEditBufferOwner**.

```
snmpset -c private crumpy vtpVlanEditBufferOwner.1 octetstring "Gerald"
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpEditControlTable.vtpEditControlEntry.
vtpVlanEditBufferOwner.1 : OCTET STRING- (ascii):    Gerald
```

5. Este ejemplo muestra cómo verificar que la tabla existe:

```
snmpwalk -c public crumpy vtpVlanEditTable
vtpVlanEditState.1.1 : INTEGER: operational
vtpVlanEditState.1.2 : INTEGER: operational
vtpVlanEditState.1.3 : INTEGER: operational
..
```

6. Este ejemplo es VLAN 11 y muestra cómo crear una fila y establecer el tipo y el nombre:

```
snmpset -c private crumpy vtpVlanEditRowStatus.1.11 integer 4
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpVlanEditTable.vtpVlanEditEntry.vtpVla
nEditRowStatus.1.11 : INTEGER: createAndGo

snmpset -c private crumpy vtpVlanEditType.1.11 integer 1
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpVlanEditTable.vtpVlanEditEntry.vtpVla
nEditType.1.11 : INTEGER: ethernet

snmpset -c private crumpy vtpVlanEditName.1.11 octetstring "test_11_gerald"
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpVlanEditTable.vtpVlanEditEntry.vtpVla
nEditName.1.11 : DISPLAY STRING- (ascii):  test_11_gerald
```

7. Establezca el **vtpVlanEditDot10Said**. Este es el número de VLAN + 100000 traducido a hexadecimal. Este ejemplo crea VLAN 11, por lo que el **vtpVlanEditDot10Said** debe ser: 11 + 100000 = 100011 -> Hex: 000186AB

```
snmpset -c private crumpy vtpVlanEditDot10Said.1.11 octetstringhex 000186AB
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpVlanEditTable.vtpVlanEdi
ntry.vtpVlanEditDot10Said.1.11 : OCTET STRING-   (hex): length = 4
     0:  00 01 86 ab -- -- -- -- -- -- -- -- -- -- -- --     ................
```

8. Cuando haya creado la VLAN 11, debe aplicar las modificaciones. Utilice el OID **vtpVlanEditOperation** de nuevo. Esta vez use el comando **Apply** para confirmar la

configuración :

```
snmpset -c private crumpy vtpVlanEditOperation.1 integer 3
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpEditControlTable.vtpEditControlEntry.
vtpVlanEditOperation.1 : INTEGER: apply
```

9. Verifique que la VLAN se haya creado correctamente. Utilice el OID **vtpVlanApplyStatus**. Verifique el proceso hasta que el estado sea: `exitoso`:

```
snmpget -c public crumpy vtpVlanApplyStatus.1
vtpVlanApplyStatus.1 : INTEGER: inProgress
snmpget -c public crumpy vtpVlanApplyStatus.1
vtpVlanApplyStatus.1 : INTEGER: inProgress
snmpget -c public crumpy vtpVlanApplyStatus.1
vtpVlanApplyStatus.1 : INTEGER: succeeded
```

10. La última acción es confirmar las modificaciones y liberar los permisos para que otros usuarios puedan agregar, modificar o eliminar VLAN de sus NMS.

```
snmpset  -c private crumpy vtpVlanEditOperation.1 integer 4
vtpVlanEditOperation.1 : INTEGER: release
```

11. Verifique que el búfer esté vacío:

```
snmpwalk -c public crumpy vtpVlanEditTable
no MIB objects contained under subtree.
```

12. Verifique que la VLAN 11 se creó en el switch con el comando CLI **show vlan** o con una **snmpwalk**:

```
snmpwalk -c public crumpy vtpVlanState
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.
1.1 : INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.
1.11 : INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.
1.48 : INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.
1.1002 : INTEGER: operational
…
```

Agregar una VLAN a un switch Cisco Catalyst con SNMP

Instrucciones de un paso

El proceso de un paso utiliza los números OID en lugar de los nombres OID como el proceso paso a paso anterior. Vea los detalles de MIB para la traducción. Este ejemplo crea VLAN 6:

```
snmpset -c private crumpy 1.3.6.1.4.1.9.9.46.1.4.1.1.1.1 integer 2
1.3.6.1.4.1.9.9.46.1.4.1.1.3.1 octetstring "gcober"

snmpset -c private gooroo 1.3.6.1.4.1.9.9.46.1.4.2.1.11.1.6 integer 4
1.3.6.1.4.1.9.9.46.1.4.2.1.3.1.6 integer 1 1.3.6.1.4.1.9.9.46.1.4.2.1.4.1.6 octetstring "vlan6"
1.3.6.1.4.1.9.9.46.1.4.2.1.6.1.6 octetstringhex 000186A6 1.3.6.1.4.1.9.9.46.1.4.1.1.1.1 integer
3

snmpset -c private gooroo 1.3.6.1.4.1.9.9.46.1.4.1.1.1.1 integer 4
```

```
snmpwalk -c public crumpy 1.3.6.1.4.1.9.9.46.1.3.1.1.2
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1.1 :
INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1.6 :
INTEGER: operational
cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1.11 :
INTEGER: operational
```

Nota: Ciertas versiones SNMP requieren que utilice un (.) antes del OID en los comandos SNMP SET.

Eliminación de una VLAN de un Cisco Catalyst Switch con SNMP

Step-by-Step Instructions

En este ejemplo, la VLAN 48 se elimina del switch. Consulte Agregar una VLAN a un Cisco Catalyst con SNMP para obtener más información. La diferencia entre esta sección donde se elimina una VLAN y la que se agrega una VLAN es que se utiliza el comando **destruir** en lugar del comando **CreateAndGo** para el **vtpVlanEditRowStatus**:

1. Ejecute el comando para eliminar la VLAN 48:
   ```
   snmpset -c private crumpy vtpVlanEditOperation.1 integer 2
   cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpEditControlTable.vtpEditControlEntry.
   vtpVlanEditOperation.1 : INTEGER: copy
   snmpset -c private crumpy vtpVlanEditRowStatus.1.48 integer 6
   cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanEdit.vtpVlanEditTable.vtpVlanEditEntry.vtpVla
   nEditRowStatus.1.48 : INTEGER: destroy
   ```

2. Para verificar que se eliminó la VLAN 48, utilice **vtpVlanState** o **show vlan** en la CLI:
   ```
   snmpwalk -c public crumpy vtpVlanState
   cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1
   .1 : INTEGER: operational
   cisco.ciscoMgmt.ciscoVtpMIB.vtpMIBObjects.vlanInfo.vtpVlanTable.vtpVlanEntry.vtpVlanState.1
   .1002 : INTEGER: operational
   …
   ```

Agregar un puerto a una VLAN en un switch Cisco Catalyst con SNMP

Este ejemplo muestra cómo agregar un puerto Fast Ethernet 0/5 a VLAN 48.

1. Para verificar qué ifIndex Fast Eth 0/5 tiene, ejecute una **snmpwalk** de **ifDescr**:
   ```
   snmpwalk -c public crumpy ifDescr
   …
   interfaces.ifTable.ifEntry.ifDescr.6 : DISPLAY STRING- (ascii):  FastEthernet0/5
   …
   ```

2. Dado que sabe que el puerto Fast Eth 0/5 tiene un ifIndex de 6, agregue el puerto a la VLAN 48:
   ```
   snmpset -c private crumpy vmVlan.6 integer 48
   ```

```
cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembers
hipTable.vmMembershipEntry.vmVlan.6 : INTEGER: 48
```

3. Verifique que el puerto se agregó correctamente consultando el mismo OID otra vez.
   snmpget -c public crumpy vmVlan.6
   ```
   cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembers
   hipTable.vmMembershipEntry.vmVlan.6 : INTEGER: 48
   ```
   También puede verificar esto en el switch:
   ```
   crumpy#sh vlan
   VLAN Name                             Status    Ports
   ---- -------------------------------- --------- -------------------------------
   1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4,
                                                   Fa0/6, Fa0/7, Fa0/8, Fa0/9,
                                                   Fa0/10, Fa0/11, Fa0/12, Fa0/13,
                                                   Fa0/14, Fa0/15, Fa0/16, Fa0/17,
                                                   Fa0/18, Fa0/19, Fa0/20, Fa0/21,
                                                   Fa0/22, Fa0/23, Fa0/24, Gi0/1,
                                                   Gi0/2
   48   VLAN0048                         active    Fa0/5
   ```

Cómo Cambiar un Puerto de una VLAN a otra VLAN

Este ejemplo muestra cómo el puerto Fast Eth 0/3 pertenece a la VLAN 48 y cómo moverlo a la VLAN 1 (VLAN predeterminada):

1. Para verificar qué ifIndex Fast Eth 0/3 tiene, ejecute una **snmpwalk** de **ifDescr**:
   snmpwalk -c public crumpy ifDescr
   ```
   …
   interfaces.ifTable.ifEntry.ifDescr.4 : DISPLAY STRING- (ascii):  FastEthernet0/3
   …
   ```

2. Dado que sabe que el puerto Fast Eth 0/3 tiene un ifIndex de 4, puede verificar a qué VLAN pertenece actualmente el puerto:
   snmpget -c public crumpy vmVlan.4
   ```
   cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembers
   hipTable.vmMembershipEntry.vmVlan.4 : INTEGER: 48
   ```

3. El puerto pertenece a la VLAN 48.
   snmpset -c private crumpy vmVlan.4 integer 1
   ```
   cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembers
   hipTable.vmMembershipEntry.vmVlan.4 : INTEGER: 1
   ```

4. Para mover el puerto de VLAN 48 a VLAN 1, ejecute un **snmpset** de **vmVlan**.
5. Para verificar si el puerto se cambió a la otra VLAN, consulte **vmVlan** nuevamente:
   snmpget -c public crumpy vmVlan.4
   ```
   cisco.ciscoMgmt.ciscoVlanMembershipMIB.ciscoVlanMembershipMIBObjects.vmMembership.vmMembers
   hipTable.vmMembershipEntry.vmVlan.4 : INTEGER: 1
   ```

   También puede verificar esto en el propio switch :Antes del cambio:
   ```
   crumpy#sh vlan
   VLAN Name                             Status    Ports
   ---- -------------------------------- --------- -------------------------------
   1    default                          active    Fa0/1, Fa0/2, Fa0/4, Fa0/5,
                                                   Fa0/6, Fa0/7, Fa0/8, Fa0/9,
   ```

```
                                          Fa0/10, Fa0/11, Fa0/12, Fa0/13,
                                          Fa0/14, Fa0/15, Fa0/16, Fa0/17,
                                          Fa0/18, Fa0/19, Fa0/20, Fa0/21,
                                          Fa0/22, Fa0/23, Fa0/24, Gi0/1,
                                          Gi0/2
48   VLAN0048                         active    Fa0/3
```

Después del cambio:

```
crumpy#sh vlan
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4,
                                                Fa0/5, Fa0/6, Fa0/7, Fa0/8,
                                                Fa0/9, Fa0/10, Fa0/11, Fa0/12,
                                                Fa0/13, Fa0/14, Fa0/15, Fa0/16,
                                                Fa0/17, Fa0/18, Fa0/19, Fa0/20,
                                                Fa0/21, Fa0/22, Fa0/23, Fa0/24,
                                                Gi0/1, Gi0/2
48   VLAN0048                         active
```

Nota: Puede realizar otros cambios, como el nombre de VLAN, el propietario y mucho más. Consulte la MIB completa para obtener más detalles sobre el OID.

Información Relacionada

- Soporte Técnico - Cisco Systems