tices for Catalyst 4000, 5000 and 6000 Series Switch Configura

Table of Contents

Best Practices for Catalyst 4000, 5000 and 6000 Series Switch Configuration and Management

Introduction

This document discusses the implementation of Cisco Catalyst Series switches in your network, specifically the Catalyst 4000, 5000, and 6000 platforms. Configurations and commands will be discussed that assume you are running CatOS General Deployment software 5.5(7) or above. Although some design considerations are presented, this paper does not cover overall campus design.

The solutions offered here represent years of field experience from Cisco engineers working with many of our largest customers and complex networks. Consequently, this document emphasizes the kind of "real world" configurations that make networks successful, solutions that:

- Have statistically the broadest field exposure, and thus the lowest risk
- Are simple, trading some flexibility for deterministic results
- Are easy to manage and configure by network operations teams
- Promote high availability and high stability

This document is divided into four sections:

- Basic Configuration—Features used by a majority of networks such as STP and trunking
- Management Configuration—Design considerations along with system and event monitoring using SNMP, RMON, Syslog, CDP, and NTP
- Security Configuration—Passwords, port security, physical security, and authentication using TACACS+
- Configuration Checklist—Summary of suggested configuration templates

References

This paper assumes familiarity with the Catalyst 6000 Family Command Reference.

Although references to public online material for further reading will be provided throughout the document, other good foundational and educational references are:

- IOS Essentials—Layer 3 router security and routing configuration templates to complement this Layer 2 document.
- LAN Switching CCIE, Cisco Press
- Building Cisco Multilayer Switched Networks, Cisco Press
- Performance and Fault Management, Cisco Press
- Cisco Network Monitoring and Event Correlation Guidelines
- Cisco LAN Technologies Technical Assistance Center
- Gigabit Campus Network Design−−− Principles and Architecture
- Gigabit Campus Design Configuration and Recovery Analysis
- Network Management Best Practices
- Cisco SAFE: A Security Blueprint for Enterprise Networks

Basic Configuration

Features deployed when using the majority of Catalyst networks are discussed in this section.

Catalyst Control Plane Protocols

This section introduces the protocols that run between switches under normal operation. A basic understanding of them will be helpful in tackling each section.

Supervisor Traffic

Most features enabled in a Catalyst network require two or more switches to cooperate, so there must be a controlled exchange of keepalive messages, configuration parameters, and management changes. Whether these protocols are Cisco proprietary, like Cisco Discovery Protocol (CDP), or standards−based, like IEEE 802.1d (STP), all have certain elements in common when implemented on the Catalyst series.

First, let us review basic frame forwarding. User data frames originate from end systems, and their source address and destination address are not changed throughout Layer 2 switched domains. Content Addressable Memory (CAM) lookup−tables on each switch Supervisor are populated by a source address learning process and indicate which egress port should forward each frame received. If the address learning process is incomplete (the destination is unknown or the frame is destined to a broadcast or multicast address), it is forwarded (flooded) out all ports in that VLAN.

The switch must also recognize which frames are to be switched through the system and which should be directed to the switch CPU itself (also known as the Network Management Processor or NMP).

The Catalyst control plane is created using special entries in the CAM table called **system entries** to receive and direct traffic to the NMP on an internal switch port. Thus, by using protocols with well−known destination MAC addresses, control plane traffic can be separated from the data traffic. Issuing the command **show CAM system** on a switch will confirm this:

```
>show cam system

* = Static Entry. + = Permanent Entry. # = System Entry. R = Router Entry.
X = Port Security Entry
VLAN  Dest MAC/Route Des    [CoS]  Destination Ports or VCs / [Protocol Type]
----  -----------------    -----  ------------------------------------------
1     00−d0−ff−88−cb−ff  #          1/3
!−−− NMP internal port
1     01−00−0c−cc−cc−cc  #          1/3
!−−− CDP, etc.
1     01−00−0c−cc−cc−cd  #          1/3
!−−− Cisco STP
1     01−80−c2−00−00−00  #          1/3
!−−− IEEE STP
1     01−80−c2−00−00−01  #          1/3
!−−− IEEE Flow control
1     00−03−6b−51−e1−82 R#          15/1
!−−− MSFC router
...
```

Cisco has a reserved range of Ethernet MAC and protocol addresses as shown below. Each one will be covered later in this document, but a summary is presented here for convenience.

Feature	SNAP HDLC Protocol Type	Destination Multicast MAC
Port Aggregation Protocol − PAgP	0x0104	01−00−0c−cc−cc−cc
Spanning Tree PVSTP+	0x010b	01−00−0c−cc−cc−cd
VLAN Bridge	0x010c	01−00−0c−cd−cd−ce
Unidirectional Link Detection − UDLD	0x0111	01−00−0c−cc−cc−cc
Cisco Discovery − CDP	0x2000	01−00−0c−cc−cc−cc
Dynamic Trunking − DTP	0x2004	01−00−0c−cc−cc−cc
STP Uplink Fast	0x200a	01−00−0c−cd−cd−cd
IEEE Spanning Tree 802.1d	N/A − DSAP 42 SSAP 42	01−80−c2−00−00−00
Inter Switch Link − ISL	N/A	01−00−0c−00−00−00
VLAN Trunking − VTP	0x2003	01−00−0c−cc−cc−cc
IEEE Pause, 802.3x	N/A − DSAP 81 SSAP 80	01−80−C2−00−00−00>0F

The majority of Cisco control protocols use an IEEE 802.3 SNAP encapsulation, including **LLC 0xAAAA03, OUI 0x00000C,** as can be seen on a LAN analyzer trace. Other common properties of these protocols include

the following:

- These protocols assume point−to−point connectivity. Note that the deliberate use of multicast destination addresses enables two Catalysts to transparently communicate over non−Cisco switches, as devices that do not understand and intercept the frames will simply flood them. However, point−to−multipoint connections through multi−vendor environments can result in inconsistent behavior and should generally be avoided.
- These protocols terminate at Layer 3 routers; they function only within a switch domain.
- These protocols receive prioritization over user data by ingress ASIC processing and scheduling.

After introducing the control protocol destination addresses, the source address should also be described for completeness. Switch protocols use a MAC address taken from a bank of available addresses provided by an EPROM on the chassis. A **show module** command will display the address ranges available to each module when it is sourcing traffic such as STP BPDUs or ISL frames:

```
>show module

...
Mod MAC−Address(es)                        Hw     Fw          Sw
--- -------------------------------------- ------ ----------  -----------------
1   00−01−c9−da−0c−1e to 00−01−c9−da−0c−1f 2.2    6.1(3)      6.1(1d)
    00−01−c9−da−0c−1c to 00−01−c9−da−0c−1
    00−d0−ff−88−c8−00 to 00−d0−ff−88−cb−ff
!−−− MACs for sourcing traffic
...
```

VLAN 1

VLAN 1 has a special significance in Catalyst networks.

The Catalyst Supervisor always uses the **Default VLAN**, VLAN 1, to tag a number of control and management protocols when trunking, such as CDP, VTP and PAgP. All ports, including the internal sc0 interface, are configured by default to be members of VLAN 1. All trunks carry VLAN 1 by default, and in software versions before CatOS 5.4 it was not possible to block user data in VLAN 1.

Two other definitions are needed to help clarify some well−used terms in Catalyst networking:

- The **Management VLAN** is where sc0 resides; this VLAN can be changed.
- The **Native VLAN** is defined as the VLAN to which a port will return when not trunking and is the untagged VLAN on an 802.1Q trunk.

There are several good reasons to tune a network and alter the behavior of ports in VLAN 1:

- When the **diameter** of VLAN 1, like any other VLAN, gets large enough to be a risk to stability, particularly from an STP perspective, it needs to be pruned back. This is discussed in more detail in the In Band Management section of this document.
- Control plane data on VLAN 1 should be kept separate from the user data to simplify troubleshooting and maximize available CPU cycles.
- Layer 2 loops in VLAN 1 must be avoided when designing Multilayer Campus networks without STP, yet trunking is still required to the access layer if there are multiple VLANs and IP subnets. To do this, manually clear VLAN 1 from trunk ports.

In summary, it is worth noting that on trunks:

- **CDP, VTP, and PAgP** updates are always forwarded on trunks with a VLAN 1 tag. This is the case even if VLAN 1 has been cleared from the trunks and is not the native VLAN. Clearing VLAN 1 for user data has no impact on control plane traffic that is still sent using VLAN 1.
- **802.1Q IEEE BPDUs** are forwarded untagged on the "Common Spanning Tree" VLAN 1 for interoperability with other vendors, unless VLAN 1 has been cleared from the trunk. **Cisco Per−VLAN Spanning Tree (PVST+) BPDUs** are sent and tagged for all other VLANs. See the Spanning Tree Protocol section in this document for more details.
- **DTP** updates are forwarded tagged on VLAN 1 ISL trunks, but use the native VLAN on 802.1Q trunks.

VLAN Trunking Protocol (VTP)

Before creating VLANs, determine the VTP mode to be used in the network. VTP enables VLAN configuration changes to be made centrally on one or more switches. Those changes automatically propagate to all other switches in the domain.

Operational Overview

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network−wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause a number of problems, such as duplicate VLAN names, incorrect VLAN−type specifications, and security violations. The VLAN database is a binary file and is stored in NVRAM on VTP servers separately from the configuration file.

The VTP protocol communicates between switches using an Ethernet destination multicast MAC address (**01−00−0c−cc−cc−cc**) and SNAP HDLC protocol type Ox2003. It does not work over non−trunk ports (VTP is a payload of ISL or 802.1Q), so messages cannot be sent until DTP has brought the trunk online.

Message types include "summary advertisements" every five minutes, "subset advertisements and request advertisements" when there are changes, and "Joins" when VTP pruning is enabled. The VTP configuration revision number is incremented by one with every change on a server, which then propagates the new table across the domain.

If a VLAN is deleted, ports that were once a member of that VLAN are placed in an inactive state. Similarly, if a switch in client mode is unable to receive the VTP VLAN table at boot−up (either from a VTP server or another VTP client), all ports in VLANs other than the default VLAN 1 will be deactivated.

Feature	Server	Client	Transparent
Source VTP Messages	Yes	Yes	No
Listen to VTP messages	Yes	Yes	No
Create VLANs	Yes	No	Yes (locally significant only)
Remember VLANs	Yes	No	Yes (locally significant only)

In VTP transparent mode, VTP updates are ignored (the VTP multicast MAC address is removed from the system CAM that is normally used to pick up control frames and direct them to the Supervisor). As the protocol uses a multicast address, a switch in transparent mode (or another vendor's switch) will simply flood the frame to other Cisco switches in the domain.

Here is a summary of the initial configuration:

Feature	Default Value
VTP Domain Name	Null
VTP mode	Server
VTP version 2	Disabled
VTP password	None
VTP Pruning	Disabled

VTP version 2 includes the following functional flexibility, but is not interoperable with VTP version 1:

- Token Ring support
- Unrecognized VTP information support—Switches will now propagate values they cannot parse.
- Version−dependent transparent mode—Transparent mode no longer checks domain name; this enables support of more than one domain across a transparent domain.
- Version number propagation—If VTPv2 is possible on all switches, then all can be enabled by configuring a single switch.

For more information see: Understanding and Configuring VLAN Trunk Protocol (VTP).

Recommendation

There is no specific recommendation on whether to use VTP client/server modes or VTP transparent mode. Some customers prefer the ease of management of VTP Client/Server mode despite some considerations noted below. The recommendation is to have two "server mode" switches in each domain for redundancy, typically the two distribution−layer switches. The rest of the switches in the domain should be set to "client mode."

There are pros and cons to VTP's ability to make changes easily on a network, and many enterprises prefer a cautious approach of using VTP transparent mode for the following reasons:

- It encourages good change control practice, as the requirement to modify a VLAN on a switch or trunk port has to be considered one switch at a time.
- It limits the risk of an administrator error, such as deleting a VLAN accidentally and thus impacting the entire domain.
- There is no risk from a new switch being introduced into the network with a higher VTP revision number and overwriting the entire domain's VLAN configuration.
- It encourages VLANs to be pruned from trunks running to switches that do not have ports in that VLAN, thus making frame flooding more bandwidth−efficient. Manual pruning also has the benefit of reducing the spanning tree diameter (see DTP section).
- The extended VLAN range in CatOS 6.x, numbers 1025−4094, can only be configured in this way.
- VTP Transparent mode is supported in Campus Manager 3.1, part of Cisco Works 2000. The old restriction of needing at least one server in a VTP domain has been removed

Sample VTP Commands	Comments
set vtp domain <*name*> password <*x*>	CDP checks names to help check for miscabling between domains. A simple password is a helpful precaution against unintentional changes. Beware of case−sensitive names or spaces if pasting.

set vtp mode transparent	
set vlan <*vlan number*> name <*name*>	Per switch that has ports in the VLAN.
set trunk <*mod/port*> <*vlan range*>	Enables trunks to carry VLANs where needed − default is all VLANs.
clear trunk <*mod/port*> <*vlan range*>	Limit STP diameter by manual pruning, such as on trunks from distribution layer to access layer, where the VLAN does not exist.

Note that specifying VLANs using the **set** command will only add VLANs, not clear them. For example, **set trunk x/y 1−10** does not set the allowed list to just VLANs 1−10. Enter **clear trunk x/y 11−1005** to achieve the desired result.

Although token ring switching is outside the scope of this document, it is worth noting that VTP transparent mode is not recommended for TR−ISL networks. The basis for token ring switching is that the whole domain forms a single distributed multi−port bridge; therefore, every switch must have the same VLAN information.

Other Options

VTPv2 is a requirement in token ring environments, where Client/Server Mode is highly recommended.

The benefits of pruning VLANs to reduce unnecessary frame flooding have been advocated in the previous section. The **set vtp pruning enable** command will prune VLANs automatically, stopping the inefficient flooding of frames where they are not needed. Note that unlike manual VLAN pruning, automatic pruning does not limit the spanning tree diameter.

In CatOS 6.x, Catalyst switches support 4096 VLANs in accordance with the IEEE 802.1Q standard. These VLANs are organized into three ranges, only some of which are propagated to other switches in the network with VTP:

- Normal−range VLANs: 1−1001.
- Extended−range VLANs: 1025−4094 NOT propagated by VTP.
- Reserved−range VLANs: 0, 1002−1024, 4095

The IEEE has produced a standards−based architecture to accomplish similar results as VTP. As a member of the 802.1Q Generic Attribute Registration Protocol (GARP), the Generic VLAN Registration Protocol (GVRP) will allow VLAN management interoperability between vendors, but is outside the scope of this document.

Finally, it is worth noting that CatOS 7.x introduces the possibility to set VTP to “off” mode, a mode very similar to “transparent,” but the switch does not forward VTP frames. This may be useful in some designs when trunking to switches outside of your administrative control.

Auto−Negotiation

Ethernet / Fast Ethernet

Auto−negotiation is an optional function of the IEEE Fast Ethernet standard (802.3u) that enables devices to automatically exchange information over a link about **speed** and **duplex** abilities. Auto−negotiation operates at Layer 1 and targets access layer ports where **transient users** such as PCs connect to the network.

Operational Overview

The most common cause of performance issues on 10/100 Mbps Ethernet links is when one port on the link is operating at half duplex while the other is at full duplex. This occasionally happens when one or both ports on a link are reset and the auto−negotiation process does not result in both link partners having the same configuration. It also happens when administrators reconfigure one side of a link and forget to reconfigure the other side. The typical symptoms of this are increasing FCS, CRC, alignment, or runt counters on the switch.

Auto−negotiation is discussed in detail in the following documents with explanations of how auto−negotiation works and configuration options.

- Configuring and Troubleshooting Ethernet 10/100Mb Half/Full Duplex Auto−Negotiation
- Troubleshooting Cisco Catalyst Switches to Network Interface Card (NIC) Compatibility Issues

A common misconception about auto−negotiation is that it is possible to manually configure one link partner for 100 Mbps full duplex and auto−negotiate to full duplex with the other link partner. In fact, attempting to do this will result in a duplex mismatch. This is a consequence of one link partner auto−negotiating, not seeing any auto−negotiation parameters from the other link partner, and defaulting to half−duplex.

Most Catalyst Ethernet modules support 10/100 Mbps and half/full duplex, but the **show port capabilities <*mod/port*>** command will confirm this.

FEFI

Far−End−Fault−Indication (FEFI) protects 100BaseFX (fiber) and Gigabit interfaces, while auto−negotiation protects 100BaseTX (copper) against physical−layer/signaling related faults.

A **far end fault** is an error in the link that one station can detect while the other cannot, such as a disconnected TX−wire. In this example, the sending station would still be receiving valid data and detect that the link is good via the link−integrity−monitor; however, it will not be able to detect that its transmission is not being received by the other station. A 100BaseFX station that detects such a remote fault may modify its transmitted IDLE stream to send a special bit−pattern (referred to as the FEFI IDLE pattern) to inform the neighbor of the remote fault; the FEFI−IDLE pattern subsequently triggers a shutdown of the remote port (ErrDisable). See the UDLD section in this document for further information on fault protection.

FEFI is supported by the following hardware/modules:

- Catalyst 5000: WS−X5201R, WS−X5305, WS−X5236, WS−X5237, WS−U5538, and WS−U5539
- Catalyst 6000 and 4000: All 100BaseFX modules and GE modules

Recommendation

Whether to configure auto−negotiation on 10/100 links or to "hard code" speed and duplex ultimately depends on the type of link partner or end device you have connected to a Catalyst switch port. Auto−negotiation between end devices and Catalyst switches generally works well, and Catalyst switches are compliant with the IEEE 802.3u specification. However, problems may result when NIC or vendor switches do not conform exactly. Hardware incompatibility and other issues may also exist as a result of vendor−specific advanced features, such as auto−polarity or cabling integrity, that are not described in the IEEE 802.3u specification for 10/100 Mbps auto−negotiation. An example is given in the following field notice:

Field Notice: Performance Issue with Intel Pro/1000T NICs connecting to CAT4K/6K

Therefore, it is worth anticipating there will be some situations that require host, port speed, and duplex to be set**.** In general, follow these basic troubleshooting steps:

- Make sure that either auto−negotiation is configured on both sides of the link or hard coding is configured on both sides.
- Check the CatOS release notes for common caveats.
- Verify the version of NIC driver or operating system you are running, as the latest driver or patch is often required.

As a rule, try using auto−negotiation first for any type of link partner. There are obvious benefits to configuring auto−negotiation for transient devices like laptops. Auto−negotiation should also work well with non−transient devices like servers and fixed workstations or from switch−to−switch and switch−to−router. However, for some of the reasons mentioned above, negotiation issues may arise. In these cases, follow the basic troubleshooting steps as outlined in the TAC links provided.

If the port speed is set to auto on a 10/100 Mbps Ethernet port, both speed and duplex are auto−negotiated. Use the following command to set the port to auto:

```
set port speed <port range> auto
!−−− this is the default
```

If hard coding the port, use the following configuration commands:

```
set port speed <port range> <10 | 100 >

set port duplex <port range> <full | half>
```

Other Options

When no auto−negotiation is used between switches, Layer 1 fault indication can also be lost for certain problems. It is helpful to use Layer 2 protocols to augment failure detection, like aggressive UDLD (see later section).

Gigabit Ethernet

Gigabit Ethernet has an auto−negotiation procedure (IEEE 802.3z) that is more extensive than that for 10/100 Mbps Ethernet and is used to exchange flow−control parameters, remote fault information, and duplex information (even though Catalyst series Gigabit Ethernet ports only support full−duplex mode). Note that 802.3z has been superseded by IEEE 802.3:2000 specs (see IEEE Standards On Line LAN/MAN Standards Subscription: Archives).

Operational Overview

Gigabit Ethernet port negotiation is enabled by default, and the ports on both ends of a Gigabit Ethernet link must have the same setting. Unlike Fast Ethernet, the GE link will not come up if the ports at each end of the link are set inconsistently (the exchanged parameters are different). Also unlike Fast Ethernet, Gigabit Ethernet does not negotiate port speed, and you cannot disable auto−negotiation using the **set port speed** command.

For example, assume that there are two devices, A and B, and that each device can have auto−negotiation enabled or disabled. Below is a list of possible configurations and their respective link states:

Negotiation	B Enabled	B Disabled

A Enabled	**Up at both sides**	A Down, B Up.
A Disabled	A Up, B Down	Down at both sides

Recommendation

Enabling auto−negotiation is much more critical in a Gigabit Ethernet environment than in a 10/100 environment. In fact, auto−negotiation should only be disabled on switch ports that attach to devices not capable of supporting negotiation or where connectivity issues arise from interoperability issues. Cisco recommends that Gigabit negotiation be enabled (default) on all switch−to−switch links and generally all GE devices. Use the following command to enable auto−negotiation:

```
set port negotiation <port range> enable
!−−− this is the default
```

One known exception is when connecting to a Gigabit Switch Router (GSR) running IOS prior to version 12.0(10)S, the release that added flow control and auto−negotiation. In this case, turn off those two features, or the switch port will report "not connected" and the GSR will report errors. Here is a sample command sequence:

```
set port flowcontrol receive <port range> off
set port flowcontrol send <port range> off
set port negotiation <port range> disable
```

Switch−to−server connections must be looked at on a case−by−case basis. Cisco customers have encountered issues with Gigabit negotiation on Sun, HP, and IBM servers.

Other Options

Flow−Control is an optional part of the 802.3x specification and must be negotiated if used. Devices may or may not be capable of sending and/or responding to a PAUSE frame (**well known MAC 01−80−C2−00−00−00 0F**), and they may not agree to the flow−control request of the far−end neighbor. A port with an input buffer that is filling up sends a PAUSE frame to its link partner, which stops transmitting, holding any additional frames in the link partner's output buffers. This does not solve any steady−state over−subscription problem, but effectively makes the input buffer larger by some fraction of the partner's output buffer during bursts.

This feature is best used on links between access−ports and end hosts, where the host "output buffer" is potentially as large as their virtual memory. Switch−to−switch use has limited benefits.

Use the following commands to control this on the switch ports:

```
set port flowcontrol <mod/port> <receive | send> <off |on | desired>

>show port flowcontrol

 Port  Send FlowControl    Receive FlowControl   RxPause TxPause
       admin    oper       admin    oper
−−−−−  −−−−−−−− −−−−−−−−   −−−−−−−− −−−−−−−−     −−−−−−− −−−−−−−
 6/1   off      off        on       on           0       0
 6/2   off      off        on       on           0       0
 6/3   off      off        on       on           0       0
```

Note: All Catalyst modules will respond to a PAUSE frame if negotiated. Some modules (eg, WS−X5410, WS−X4306) will never send pause frames even if they negotiate to do so, as they are non−blocking.

Dynamic Trunking Protocol (DTP)

Encapsulation Type

Trunks extend VLANs between devices by temporarily identifying and tagging (link−local) the original Ethernet frames, thus enabling them to be multiplexed over a single link. This also ensures the separate VLAN broadcast and security domains are maintained between switches. Content Addressable Memory (CAM) tables maintain the frame−to−VLAN mapping inside the switches.

Trunking is supported on several Layer 2 media, including ATM LANE, FDDI 802.10, and Ethernet, although only the latter will be presented here.

ISL Operational Overview

Cisco's proprietary identification or tagging scheme, Inter Switch Link (ISL), has been in use for many years, and now the 802.1Q IEEE standard is also available.

By totally encapsulating the original frame in a "two−level" tagging scheme, ISL is effectively a tunneling protocol and has the additional benefit of carrying non−Ethernet frames. It adds a 26−byte header and 4−byte Frame Check Sequence (FCS) to the standard Ethernet frame − the larger Ethernet frames are expected and handled by ports configured to be trunks. ISL supports 1024 VLANs.

Frame format − ISL tag shaded:

40 Bits	4 Bits	4 Bits	48 Bits	16 Bits	24 Bits	24 Bits	15 Bits	1 Bit	16 Bits	16 Bits	Variable length	32 Bi
Dest. Addr	Type	USER	SA	LEN	SNAP LLC	HSA	VLAN	BPDU	INDEX	Reserve	Encapsulated Frame	FCS
01−00−0c−00−00					AAAA03	00000C						

For more information, see the ISL Functional Specification.

802.1Q Operational Overview

The IEEE 802.1Q standard specifies much more than encapsulation types, including spanning tree enhancements, GARP (see VTP section), and 802.1p quality of service (QoS) tagging.

The 802.1Q frame format preserves the original Ethernet Source Address and Destination Address, yet switches must now expect "baby−giant" frames to be received, even on access ports where hosts may use tagging to express 802.1p user priority for QoS signaling. The tag is 4 bytes, so 802.1Q Ethernet v2 frames are 1522 bytes, an IEEE 802.3ac working group achievement. Finally, 802.1Q supports numbering space for 4096 VLANs.

All data frames transmitted and received are 802.1Q−tagged except for those on the native VLAN (there is an implicit tag based on the ingress switch port configuration). Frames on the native VLAN are always transmitted untagged and normally received untagged but may also be received tagged.

For more details, see VLAN Standardization via IEEE 802.10 or Get IEEE 802 .

802.1Q/801.1p frame format:

Tag Header

		TPID	TCI					
48 bits	48 bits	**16 bits**	**3 bits**	**1 bit**	**12 bits**	16 bits	Variable length	32 bits
DA	SA	**TPID**	**Priority**	**CFI**	**VLAN ID**	Length/ Type	Data with PAD	FCS
		0x8100	**0 − 7**	**0−1**	**0−4095**			

Recommendation

As all newer hardware supports 802.1Q (and some only supports 802.1Q, such as the Catalyst 4000 series and CSS 11000), Cisco recommends that all new implementations follow the IEEE 802.1Q standard and older networks gradually migrate from ISL.

The IEEE standard will allow vendor interoperability. This will be advantageous in all Cisco environments as new host 802.1p capable NICs and devices become available. Although both ISL and 802.1Q implementations are mature, the IEEE standard will ultimately have greater field exposure and greater third party support, such as network analyzer support. The lower encapsulation overhead of 802.1Q compared to ISL is a minor point in 802.1Q's favor as well.

As the encapsulation type is negotiated between switches using DTP, with ISL chosen as the winner by default if both ends support it, it is necessary to specify "dot1q" with the following command:

```
set trunk <mod/port> <mode> dot1q
```

If VLAN 1 is cleared from a trunk, as discussed in the In Band Management section, although no user data is transmitted or received, the Network Management Processor (NMP) continues to pass control protocols such as CDP and VTP on VLAN 1.

Also, as discussed in the VLAN 1 section, CDP, VTP, and PAgP packets are always sent on VLAN 1 when trunking. When using dot1Q encapsulation, these control frames will be tagged with VLAN 1 if the switch's native VLAN has been changed. If dot1Q trunking to a router is enabled and the native VLAN has been changed on the switch, a sub−interface in VLAN 1 is needed to receive the tagged CDP frames and provide CDP neighbor visibility on the router.

Note: There is a potential security consideration with dot1Q caused by the implicit tagging of the native VLAN, as it may be possible to send frames from one VLAN to another without a router: See Are there Vulnerabilites in VLAN Implementations? for further details. The workaround is to use a VLAN ID for the trunk's native VLAN that is not used for end user access. The majority of Cisco customers achieve this simply by leaving VLAN 1 as the native VLAN on a trunk and assigning access ports to VLANs other than VLAN 1.

Trunking Mode

DTP is the second generation of DISL (Dynamic ISL) and exists to ensure that the different parameters involved in sending ISL or 802.1Q frames, such as the configured encapsulation type, native VLAN, and hardware capability, are agreed upon by the switches at either end of a trunk. This also helps protect against non−trunk ports flooding tagged frames, a potentially serious security risk, by ensuring that ports and their neighbors are in consistent states.

Operational Overview

DTP is a Layer 2 protocol that negotiates configuration parameters between a switch port and its neighbor. It uses another multicast MAC address (**01−00−0c−cc−cc−cc**) and a SNAP protocol type of **0x2004**. Here is a summary of the configuration modes:

Mode	Function	DTP Frames Transmitted	Final state (Local port)
Auto(Default)	Makes the port willing to convert the link to a trunk. The port becomes a trunk port if the neighboring port is set to on or desirable mode.	Yes, periodic	Trunking
On	Puts the port into permanent trunking mode and negotiates to convert the link into a trunk. The port becomes a trunk port even if the neighboring port does not agree to the change.	Yes, periodic	Trunking, **unconditionally**
Nonegotiate	Puts the port into permanent trunking mode but prevents the port from generating DTP frames. You must configure the neighboring port manually as a trunk port to establish a trunk link. This is useful for devices that do not support DTP.	No	Trunking, **unconditionally**
Desirable	Makes the port actively attempt to convert the link to a trunk link. The port becomes a trunk port if the neighboring port is set to on, desirable, or auto mode.	Yes, periodic	It will end up in trunking state only if the remote mode is on, auto, or desirable.
Off			Non−trunking

	Puts the port into permanent non−trunking mode and negotiates to convert the link into a non−trunk link. The port becomes a non−trunk port even if the neighboring port does not agree to the change.	No in steady state, but will transmit "informs" to speed up remote end detection after change from "on."	

Here are some highlights of the protocol:

- DTP assumes a point−to−point connection, and Cisco devices will only support 802.1Q trunk ports that are point−to−point.
- During DTP negotiation, the ports will not participate in STP. Only after the port becomes one of the three DTP types (Access, ISL, or 802.1Q) will the port be added to STP. (Otherwise PAgP, if configured, is the next process to run before the port participates in STP.)
- If the port is trunking in ISL mode, DTP packets are sent out on VLAN 1, otherwise (for 802.1Q trunking or non−trunking ports) they are sent out on the native VLAN.
- In desirable mode, DTP packets transfer the **VTP domain name** (which must match for a negotiated trunk to come up), plus trunk configuration and **admin status**.
- Messages are sent every second during negotiation and every 30 seconds after that.
- Be sure to understand that modes "on," "nonegotiate," and "off" explicitly specify in which state the port will end up. A bad configuration can lead to a dangerous/inconsistent state where one side is trunking and the other is not.
- A port in "on," "auto," or "desirable" mode sends DTP frames periodically. If a port in auto or desirable mode does not see a DTP packet in five minutes, it will be set to non−trunk.

See ISL Trunking on Catalyst 5000 and 6000 Family Switches for more ISL details and Trunking Between Catalyst 4000, 5000, and 6000 Family Switches Using 802.1Q Encapsulation for 802.1Q details.

Recommendation

Cisco recommends an explicit trunk configuration of "desirable" at both ends. In this mode, network operators can trust syslog and command line status messages that a port is up and trunking, unlike "on" mode, which can make a port appear up even though the neighbor is misconfigured. In addition, "desirable" mode trunk provides stability in situations where one side of the link cannot become a trunk or drops trunk state. Use the following command to set desirable mode:

```
set trunk <mod/port> desirable <ISL | dot1q>
```

Note: Set "trunk off" on all non−trunk ports. This helps eliminate wasted negotiation time when bringing host ports up. This command is also executed when the set port host command is used − see the STP section. Use the following command to disable trunk on a range of ports:

```
set trunk <port range> off
!−−− Where ports not trunking, also part of the set port host command
```

Other Options

Another common customer configuration uses "desirable" mode only at the distribution layer and the simplest default configuration (auto mode) at the access layer.

Some switches, such as a Catalyst 2900−XL, IOS routers, or other vendors, do not currently support trunk negotiation via DTP, so "nonegotiate" mode on Catalyst 4/5/6000s can be used to set a port to trunk unconditionally with these devices, which may help standardize on a common setting across the campus. Use the following command to set nonegotiate mode:

```
set trunk <mod/port> nonegotiate <ISL | dot1q>
```

The reason Cisco recommends "nonegotiate" when connecting to an IOS router is that when performing bridging, some DTP frames received from "on" mode may get back into the trunk port. Upon reception of the DTP frame, the switch port will try to renegotiate (meaning bring the trunk down and up) unnecessarily. If "nonegotiate" is enabled, the switch will not send DTP frames.

Spanning Tree Protocol (STP)

Basic Considerations

Spanning tree maintains a loop−free Layer 2 environment in redundant switched and bridged networks. Without STP, frames would loop and/or multiply indefinitely, causing a network meltdown as all devices in the broadcast domain would be interrupted continuously by high traffic.

Although in some respects STP is a mature protocol initially developed for slow software−based bridge specifications (IEEE 802.1d), it can be complex to implement well in large switched networks with many VLANs, many switches in a domain, multi−vendor support, and newer IEEE enhancements.

For future reference, CatOS 6.x continues to take on new STP development, such as multiple instance spanning tree (MISTP), loop−guard, root−guards, and BPDU arrival time skew detection. In addition, further standardized protocols are available in CatOS 7.x, such as IEEE 802.1s shared spanning tree and IEEE 802.1w rapid convergence spanning tree.

Operational Overview

The root bridge election per VLAN is won by the switch with the lowest root bridge identifier (BID) − the bridge priority combined with the switch MAC address.

Initially, Bridge Protocol Data Units (BPDUs) are sent from all switches, containing the BID of each switch and the path cost to reach that switch; this enables the root bridge and the lowest−cost path to the root to be determined. Additional configuration parameters carried in BPDUs from the root override those that are locally configured so that the whole network uses consistent timers.

The topology then converges through the following steps:

1. A single root bridge is elected for the entire spanning tree domain
2. One root port (facing the root bridge) is elected on every non−root bridge
3. A designated port is elected for BPDU forwarding on every segment
4. Non−designated ports become blocking.

For more information see Configuring Spanning Tree.

Basic Timer Defaults (seconds)	Name	Function
2	Hello	Controls sending of BPDUs.

15	Forward Delay (Fwddelay)	Controls how long a port spends in listening and learning state and influences the topology change process (see below).
20	Maxage	Controls how long the switch will maintain the current topology before looking for an alternative path. After Maxage seconds, a BPDU is considered stale and the switch looks for a new root port from the pool of blocking ports. If no blocked port is available, it will claim to be the root itself on the designated ports.

Port States	Meaning	Default timing to next state
Disabled	Administratively down	N/A
Blocking	Receiving BPDUs and stopping user data.	Monitor reception of BPDUs. Wait 20 seconds for Maxage expiration or immediate change if direct/local link failure detected.
Listening	Sending or receiving BPDUs to check whether return to blocking needed.	Fwddelay timer (wait 15 seconds)
Learning	Building topology/CAM table	Fwddelay timer (wait 15 seconds)
Forwarding	Sending/receiving data	
	Total basic topology change:	**20 + 2 (15) = 50 seconds if waiting for Maxage to expire, or 30 seconds for direct link failure**

There are two types of BPDUs used in spanning tree: Configuration BPDUs and Topology Change Notification (TCN) BPDUs.

Configuration BPDU Flow

Configuration BPDUs are sourced every Hello−interval from every port on the root bridge and subsequently flow to all leaf switches to maintain the state of the spanning tree. In steady state, BPDU flow is unidirectional: root ports and blocking ports only receive configuration BPDUs, while designated ports only send configuration BPDUs.

For every BPDU received by a switch from the root, a new one is processed by the Catalyst's central NMP and sent out containing the root's information. In other words, if the root bridge is lost or all paths to the root bridge are lost, then BPDUs stop being received (until the Maxage timer starts re−election).

Topology Change Notification (TCN) BPDU Flow

Topology Change BPDUs are sourced from leaf switches and flow towards the root bridge when a topology change has been detected in the spanning tree. Root ports only send TCNs and designated ports only receive TCNs.

The TCN BPDU travels towards the root ridge and is acknowledged at each step, so this is a reliable mechanism. Once it arrives at the root bridge, the root bridge alerts the entire domain that a change has occurred by sourcing Configuration BPDUs with the TCN flag set for **Maxage + Fwddelay** time (35 seconds by default). This causes all switches to change their normal CAM aging time from five minutes (by default) to the interval specified by **Fwddelay** (15 seconds by default). See Understanding Spanning−Tree Protocol Topology Changes for more details.

Spanning Tree Modes

There are three different ways to correlate VLANs with Spanning Tree:

- A single spanning tree for all VLANs, or Mono Spanning Tree, such as IEEE 802.1Q.
- A spanning tree per VLAN, or Shared Spanning Tree, such as Cisco PVST.
- A spanning tree per set of VLANs, or Multiple Spanning Tree, such as Cisco MISTP, IEEE 802.1s.

A Mono Spanning Tree for all VLANs allows only one active topology and therefore no load balancing. A STP blocked port will be blocking for all VLANs and will carry no data.

One Spanning Tree per VLAN allows load balancing but requires more BPDU CPU processing as the number of VLANs increases. The CatOS release notes provide guidance on the number of logical ports recommended in the spanning tree per switch. For example, the Catalyst 6000 Supervisor1 formula is:

number of ports + (number of trunks * number of VLANs on trunks) < 4000

Cisco MISTP and the new 802.1s standard are the way forward, allowing, for example, the definition of just two active STP instances/topologies and the mapping of all VLANs to either of these two trees. This technique allows STP to scale to many thousands of VLANs while enabling load balancing.

BPDU Formats

In order to support the IEEE 802.1Q standard, Cisco's existing STP implementation was extended to become PVST+ by adding support for tunneling across an IEEE 802.1Q Mono Spanning Tree region. PVST+ is therefore compatible with both IEEE 802.1Q's MST and Cisco PVST protocols without requiring extra commands or configuration. In addition, PVST+ adds verification mechanisms to ensure that there is no configuration inconsistency of port trunking and VLAN IDs across switches.

Here are some operational highlights of the PVST+ protocol:

- PVST+ interoperates with 802.1Q Mono Spanning Tree via the so−called Common Spanning Tree (CST) over an 802.1Q trunk. The CST is always on VLAN 1, so this VLAN needs to be enabled on the trunk to interoperate with other vendors. CST BPDUs are transmitted, always untagged, to the IEEE Standard Bridge−Group (MAC Address 01−80−c2−00−00−00, DSAP 42, SSAP 42). For completeness of description, a parallel set of BPDUs are also transmitted to the Cisco Shared Spanning Tree MAC address for VLAN 1.
- PVST+ tunnels PVST BPDUs across 802.1Q VLAN regions as multicast data. Cisco's Shared Spanning Tree BPDUs are transmitted to MAC address 01−00−0c−cc−cc−cd (SNAP HDLC protocol

type 0x010b) for each VLAN on a trunk. BPDUs are untagged on the native VLAN and tagged for all other VLANs.
- PVST+ checks port and VLAN inconsistencies. PVST+ blocks those ports that receive inconsistent BPDUs in order to prevent forwarding loops. It also notifies users via syslog messages about any configuration mismatch.
- PVST+ is backward−compatible with existing Cisco switches running PVST on ISL trunks. ISL−encapsulated BPDUs are still transmitted or received using the IEEE MAC address. In other words, each BPDU type is link−local − there are no translation issues.

Recommendation

All Catalyst switches have STP enabled by default; this is recommended even if a design is chosen that does not include Layer 2 loops, so that STP is not "enabled" in the sense that it is actively maintaining a blocked port.

```
set spantree enable all
!−−− this is the default
```

Cisco recommends leaving STP enabled for these reasons:

- If there is a loop (induced by mispatching, bad cable, etc.), STP will prevent detrimental effects to the network caused by multicast and broadcast data.
- Protection against an EtherChannel breaking down.
- Most networks are configured with STP, giving it maximum field exposure. More exposure generally equates to stable code.
- Protection against dual attached NICs misbehaving (or bridging enabled on servers).
- The software for many protocols (such as PAgP, IGMP snooping, and trunking) is closely related to STP, so running without it may lead to undesirable results.
- During a reported network disruption, Cisco engineers will most likely suggest that disabled STP will be at the center of the fault if at all conceivable.

Do not change timers, as this may adversely affect stability. The majority of networks deployed are not tuned. The "simple" STP timers accessible via the command line, such as Hello−interval and Maxage, are themselves comprised of a complex set of other assumed and intrinsic timers, so it is difficult to tune timers and consider all the ramifications. Moreover, there is the danger of undermining UDLD protection (see later section).

Ideally, keep user traffic off the management VLAN. Especially with older Catalyst switch processors, it is best to avoid problems with STP by keeping the management VLAN separate from user data. One misbehaving end station could potentially keep the Supervisor processor so busy with broadcast packets that it can miss one or more BPDUs. Newer switches with more powerful CPUs and throttling controls are relieving this consideration, however. See the In Band Management section for more details.

Do not over−design redundancy. This might lead to a troubleshooting nightmare − too many blocking ports will adversely affect long−term stability. **Keep the total SPT diameter under seven hops.** Try to design to the Cisco "multilayer" model, with its smaller switched domains, STP "triangles," and deterministic blocked ports (as explained in Gigabit Campus Network Design−−− Principles and Architecture), wherever possible.

Influence and know where Root functionality and blocked ports reside, and document them on the topology diagram. The blocked ports are where STP troubleshooting begins − what made them change from blocking to forwarding is often the key part of root cause analysis. **Choose the distribution and core layers as the location of Root/Secondary Root**, since these are considered the most stable parts of the network. Check for optimal Layer 3 and HSRP overlay with Layer 2 data−forwarding paths. The following command is

a macro that configures the bridge priority; "root" sets it much lower than the default (32768), while "root secondary" sets it reasonably lower than the default:

```
set spantree root <secondary> <vlan range>
```

Note that this macro sets the root priority to be either 8192 (by default), the current root priority minus 1 (if another root bridge is known), or the current root priority (if its MAC address is lower then the current root).

Prune unnecessary VLANs off trunk−ports (a bi−directional exercise). This will limit the diameter of STP and NMP processing overhead on portions of the network where certain VLANs are not required. VTP automatic pruning does not remove STP from a trunk − see the section on VTP. The default VLAN 1 can also be removed from trunks using CatOS 5.4 and greater.

Please see Troubleshooting Spanning−Tree Protocol and Related Design Considerations for additional information.

Other Options

Cisco has another STP protocol, called **VLAN−bridge,** that operates using a destination MAC address of **01−00−0c−cd−cd−ce** and protocol type of 0x010c.

This is most useful if there is a need to bridge non−routable or legacy protocols between VLANs without interfering with the IEEE spanning−tree instance(s) running on those VLANs. If VLAN interfaces for non−bridged traffic become blocked for Layer 2 traffic (and this could easily happen if they were participating in the same STP as IP VLANs), the overlaying Layer 3 traffic would get inadvertently pruned off as well − an unwanted side−effect. VLAN−bridge is therefore a separate instance of STP for bridged protocols, providing a separate topology that can be manipulated without affecting IP traffic.

The Cisco recommendation is to run VLAN−bridge if bridging is required between VLANs on Cisco routers such as the MSFC.

PortFast

PortFast is used to bypass normal spanning−tree operation on access ports to speed up connectivity between end−stations and the services they need to connect to after link initialization. On some protocols, like IPX/SPX, it is important to see the access port in forwarding mode immediately after the link state has gone up in order to avoid GNS problems.

For more information see Using Portfast and Other Commands to Fix Workstation Startup Connectivity Delays.

Operational Overview

PortFast skips the normal listening and learning states of STP by moving a port directly from blocking to forwarding mode after the link is known to be running. If this feature is not enabled, STP will discard all user data until it decides that the port is ready to be moved to forwarding mode. This could take up to twice the ForwardDelay time (a total of 30 seconds by default).

PortFast mode will also prevent a STP Topology Change Notification (TCN) from being generated each time a port state changes from learning to forwarding. TCNs are not a problem by themselves, but if a wave of TCNs are hitting the root bridge (typically in the morning when people turn on their PCs), it could extend convergence time unnecessarily.

STP PortFast is particularly important in both Multicast CGMP and Catalyst 5000 MultiLayer Switching (MLS) networks. TCNs in these environments can cause the static CGMP CAM table entries to be aged out, resulting in multicast packet loss until the next IGMP report, and/or flush Multi−Layer Switching cache entries that then need to be rebuilt and could result in a router CPU spike, depending on the size of the cache. (Catalyst 6000 MLS implementations and multicast entries learned from IGMP snooping are not affected.)

Recommendation

Cisco recommends that STP PortFast be enabled for all active host ports and disabled for switch−switch links and ports not in use.

Trunking and channeling should also be disabled for all host ports. Each access port is enabled by default for trunking and channeling, yet switch neighbors are not expected by design on host ports. If these protocols are left to negotiate, the subsequent delay in port activation can lead to undesirable situations in which initial packets from workstations, such as DHCP requests, are not forwarded.

CatOS 5.2 introduced a macro command, **set port host <*port range*>** that implements the following recommended configuration for access ports and will help auto−negotiation and connection performance significantly:

```
set port host <port range>
!−−− macro command for the following commands:
set spantree portfast <port range> enable
set trunk <port range> off
set port channel <port range> mode off
```

Note that PortFast does not mean that spanning−tree is not run at all on those ports: BPDUs are still sent, received, and processed.

Other Options

PortFast BPDU−Guard provides a method for preventing loops by moving a non−trunking port into an ErrDisable state when a BPDU is received on that port.

A BPDU packet should never be received on an access port configured for PortFast, since host ports should not be attached to switches. If a BPDU is observed, it indicates an invalid and possibly dangerous configuration that needs administrative action. When the BPDU−Guard feature is enabled, spanning tree shuts down PortFast−configured interfaces that receive BPDUs instead of putting them into the spanning−tree blocking state.

The command works on a per switch basis, not per port:

```
set spantree portfast bpdu−guard enable
```

The network manager is notified by an SNMP trap or syslog message if the port goes down. It is also possible to configure an automatic recovery time for ErrDisabled ports; see the UDLD section for more details. For more information, see Spanning Tree Portfast BPDU Guard Enhancement.

Note: PortFast for trunk ports was introduced in CatOS 7.x and has no effect on trunk ports in earlier releases. PortFast for trunk ports is designed to increase convergence times for Layer 3 networks. To complement this feature, CatOS 7.x also introduced the possibility of configuring PortFast BPDU−Guard on a per−port basis.

UplinkFast

UplinkFast provides fast STP convergence after a direct link failure in the network access layer. It operates without modifying the STP protocol, and its purpose is to speed up convergence time in a specific circumstance to less then 3 seconds, rather than the typical 30−second delay. For more information, see Understanding and Configuring the Cisco Uplink Fast Feature.

Operational Overview

Using Cisco's Multilayer design model at the access layer, if the forwarding uplink is lost, the blocking uplink is immediately moved to a forwarding state without waiting for listening and learning states.

An uplink group is a set of ports per VLAN that can be thought of as a root port and backup root port. Under normal conditions, the root port(s) are assuring connectivity from the access toward the root. If this primary "root−connection" fails for any reason, the backup root link kicks in immediately without having to go through typical 30 seconds of convergence delay.

Because this effectively bypasses the normal STP topology change−handling process (listening and learning), an alternate topology correction mechanism is needed to update switches in the domain that local end stations are reachable via an alternate path. Thus, the access layer switch running UplinkFast also generates frames for each MAC address in its CAM to a multicast MAC address (01−00−0c−cd−cd−cd, HDLC protocol 0x200a) to update the CAM table in all switches in the domain with the new topology.

Recommendation

Cisco recommends that UplinkFast be enabled for switches with blocked ports, typically at the access layer. Do not use on switches without the implied topology knowledge of a "backup root link" − typically distribution and core switches in Cisco's Multilayer design. It can be added without disruption to a production network. Use the following command to enable UplinkFast:

```
set spantree uplinkfast enable
```

This command will also set the **bridge priority** high to minimize the risk of this becoming a root bridge and the **port priority** high to minimize becoming a designated port, which would break the functionality. When restoring a switch that had UplinkFast enabled, the feature has to be disabled, the uplink database cleared with "clear uplink," and the bridge priorities restored manually.

Note: The "all protocols" keyword for the UplinkFast command is needed when the "protocol filtering" feature is enabled. As the CAM will record the protocol type as well as MAC and VLAN information when protocol filtering is enabled, an UplinkFast frame needs to be generated for each protocol on each MAC address. The "rate" keyword indicates the packets per second of the uplinkfast topology update frames − the default is recommended.

Backbone Fast

Backbone Fast provides rapid convergence from indirect link failures. By adding functionality to the STP protocol, convergence times can typically be reduced from the default of 50 seconds to 30 seconds.

Operational Overview

The mechanism is initiated when a root port or blocked port on a switch receives "inferior BPDUs" from its designated bridge. This can happen when a downstream switch has lost its connection to the root and starts sending its own BPDUs to elect a new root. An **inferior BPDU** identifies a switch as both the root bridge and

the designated bridge.

Under normal spanning−tree rules, the receiving switch ignores inferior BPDUs for the configured maximum aging time, 20 seconds by default. With Backbone Fast however, the switch sees the inferior BPDU as a signal that the topology may have changed, and tries to determine whether it has an alternate path to the root bridge using **Root Link Query (RLQ) BPDUs**. This protocol addition allows a switch to check whether the root is still available, moves a blocked port to forwarding in less time, and notifies the isolated switch that sent the inferior BPDU that the root is still there.

Here are some highlights of the protocol operation:

- A switch transmits the RLQ packet out the root port only (i.e. towards the root bridge).
- A switch that receives a RLQ can reply either if it is the root switch, or if it knows it has lost connection with the root. If it does not know these facts, it must forward the query out its root port.
- If a switch has lost connection to the root, it must reply in the negative to this query.
- The reply must be sent out only the port from which the query came.
- The root switch must always respond to this query with a positive reply.
- If the reply is received on a non−root port, it is discarded.

STP convergence times can therefore be reduced by up to 20 seconds, as Maxage does not need to expire.

For more information, see Understanding Spanning−Tree Protocol's Backbone Fast Feature.

Recommendation

The Cisco recommendation is to enable Backbone Fast on all switches running STP. It can be added without disruption to a production network. Use the following command to enable Backbone Fast:

```
set spantree backbonefast enable
```

Note: this global level command needs to be configured on all switches in a domain as it adds functionality to the STP protocol that all switches need to understand.

Other options

BackboneFast is not supported on 2900XLs and 3500s, so it should not be enabled if the switch domain contains these switches in addition to Catalyst 4000/5000/6000s.

Looking to the future, IEEE 802.1w rapid converge STP will provide similar reductions in STP convergence time and will be interoperable with other vendors.

EtherChannel / Port Aggregation Protocol (PAgP)

EtherChannel technologies allow the inverse multiplexing of multiple channels (up to eight on Catalyst 6000) into a single logical link. Although each platform differs from the next in implementation, it is important to understand the common requirements:

- An algorithm to statistically multiplex frames over multiple channels.
- Creation of a logical port so that a single instance of STP can be run.
- A channel management protocol described in the next section: PAgP.

Frame Multiplexing

EtherChannel encompasses a frame distribution algorithm that efficiently multiplexes frames across the component 10/100 or gigabit links. Differences in algorithms per platform arise from the capability of each type of hardware to extract frame header information in order to make the distribution decision. For example:

- The Catalyst 6000 has more recent switching hardware than the Catalyst 5000 and can read IP Layer 4 information at wire rate in order to make a more intelligent multiplexing decision than simple MAC Layer 2 information.
- The Catalyst 5000 capabilities depend on the presence of an Ethernet Bundling Chip (EBC) on the module. The command **show port capabilities <*mod/port*>** will confirm what is possible for each port.

Please refer to the following table, which illustrates the frame distribution algorithm in detail for each listed platform.

Platform	Channel Load Balancing Algorithm
Catalyst 5000 Series	A Catalyst 5000 with the necessary modules allows two to four links to be present per Fast EtherChannel, though they must be on the same module. Source and destination MAC address pairs determine the link chosen for frame forwarding. An X−OR operation is performed on the least significant two bits of the source MAC address and the destination MAC address. This operation yields one of four results: (0 0), (0 1), (1 0), or (1 1). Each of these values points to a link in the Fast EtherChannel bundle. In the case of a two−port Fast EtherChannel, only a single bit is used in the X−OR operation. Circumstances can occur where one address in the source/destination pair is a constant. For example, the destination might be a server or, even more likely, a router. In that case, statistical load balancing will be seen because the source address is always different.
Catalyst 4000 Series	Catalyst 4000 EtherChannel distributes frames across the links in a channel (on a single module) based on the low−order bits of the source and destination MAC addresses of each frame. In comparison with the Catalyst 5000, the algorithm is more involved and uses a deterministic hash of the following fields of the MAC DA (bytes 3, 5, 6), SA (bytes 3, 5, 6), ingress port, and VLAN ID. The frame distribution method is not configurable.
Catalyst 6000 Series	There are two possible hashing algorithms, depending on the Supervisor hardware. The hash is a seventeenth degree polynomial implemented in hardware that, in all cases, takes the MAC address, IP address, or IP TCP/UDP port number and applies the algorithm to generate a three bit value. This is done separately for both source and destination addresses. The results are then XOR'd to generate another three−bit value that is

	used to determine which port in the channel is used to forward the packet. Channels on the Catalyst 6000 can be formed between ports on any module and can be up to 8 ports.

The table below indicates the distribution methods supported on the various Catalyst 6000 Supervisor models and their default behavior:

Hardware	Description	Distribution Methods
WS−F6020(Layer 2 Engine)	Early Sup1	Layer 2 MAC: SA; DA; SA & DA
WS−F6020A(Layer 2 Engine)WS−F6K−PFC(Layer 3 Engine)	Later Sup1 and Sup1aSup1a/PFC1	Layer 2 MAC: SA; DA; SA &DALayer 3 IP: SA; DA; SA and DA (Default)
WS−F6K−PFC2	Sup2/PFC2Needs CatOS 6.x	Layer 2 MAC: SA; DA; SA & DALayer 3 IP: SA; DA; SA & DALayer−4 Session: S port; D port; S & D port (Default)

Note: With Layer 4 distribution, the first fragmented packet will use Layer 4 distribution. All subsequent packets will use Layer 3 distribution.

More details of EtherChannel support on other platforms and how to configure and troubleshoot them can be found at:

- Understanding and Configuring FastEtherChannel on Cisco Switching and Routing Devices
- Configuring EtherChannel Between Catalyst 4000, 5000, and 6000 Switches

Vendors that support Fast EtherChannel NICs today can be found at Fast EtherChannel® and ISL on Servers and NICs−Vendor Information.

Recommendation

Catalyst 6000 series switches perform load balancing by IP address by default, and this is recommended in CatOS 5.5, assuming that IP is the dominant protocol. Use the following command to set load balancing:

```
set port channel all distribution ip both
!--- this is the default
```

Catalyst 4000 and 5000 series frame distribution by Layer 2 MAC address is acceptable in most networks. However, the same link will be used for all traffic if there are only two main devices talking over a channel (as SMAC and DMAC are constant). This can typically be an issue for server back up and other large file transfers or for a transit segment between two routers.

Although the logical aggregate port can be managed by SNMP as a separate instance and aggregate throughput statistics gathered, Cisco still recommends managing each of the physical interfaces separately to check how the frame distribution mechanisms are working and whether statistical load balancing is being achieved.

A new command, **show channel traffic**, in CatOS 6.x can display percentage distribution statistics more easily than checking individual port counters with **show counters <*mod/port*>** or **show mac <*mod/port*>** in CatOS 5.x.

Other Options

Some possible steps to take if the relative limitations of Catalyst 4000 or Catalyst 5000 MAC based algorithms are an issue and good statistical load balancing is not being achieved:

- Point−deploy Catalyst 6000s.
- Increase bandwidth without channeling by switching, for example, from several Fast Ethernet ports to one Gigabit Ethernet port or from several Gigabit Ethernet ports to one 10Gigabit Ethernet port.
- Re−address pairs of end stations with large volume flows.
- Provision dedicated links/VLANs for high bandwidth devices.

Port Aggregation Protocol

PAgP is a management protocol that will check for parameter consistency at either end of the link and assist the channel in adapting to link failure or addition.

- PAgP requires that all ports in the channel belong to the same VLAN or are configured as trunk ports. (Because dynamic VLANs can force the change of a port into a different VLAN, they are not included in EtherChannel participation.)
- When a bundle already exists and the configuration of one port is modified (such as changing VLAN or trunking mode), all ports in the bundle are modified to match that configuration.
- PAgP does not group ports that operate at different speeds or port duplex. If speed and duplex are changed when a bundle exists, PAgP changes the port speed and duplex for all ports in the bundle.

Operational Overview

The Port Aggregation Protocol Port controls each individual physical (or logical) port to be grouped. PAgP Packets are sent using the same multicast group MAC address that is used for Cisco Discovery Protocol (CDP) packets, **01−00−0c−cc−cc−cc**, though the protocol value is 0x0104. Here is a summary of the protocol operation:

- As long as the physical port is up, PAgP packets are transmitted every second during detection and every 30 seconds in steady state.
- The protocol listens for PAgP packets that prove the physical port has a bi−directional connection to another PAgP−capable device.
- If data packets but no PAgP packets are received, it is assumed that the port is connected to a non−PAgP capable device.
- As soon as two PAgP packets have been received on a group of physical ports, it tries to form an aggregated port.
- If PAgP packets stop for a period, the PAgP state is torn down.

Normal Processing

Several concepts require defining to aid understanding of the protocol's behavior:

- **Agport**—A logical port composed of all physical ports in the same aggregation, it can be identified by its own SNMP ifIndex. Therefore, an agport does not contain non−operational ports.
- **Channel**—An aggregation satisfying the formation criteria; it therefore may contain non−operational

ports (agports are a subset of channels). Protocols including STP and VTP, but excluding CDP and DTP, run above PAgP over the agports. None of these protocols can send or receive packets until PAgP attaches their agports to one or more physical ports.

- **Group Capability**—Each physical port and agport possesses a configuration parameter called the group−capability. A physical port can be aggregated with another physical port if and only if they have the same group−capability.
- **Aggregation Procedure**—When a physical port reaches the UpData or UpPAgP states, it is attached to an appropriate agport. When it leaves either of those states for another state, it is detached from the agport.

Definitions of the states are given below, followed by the creation procedure.

State	Meaning
UpData	No PAgP packets have been received. PAgP packets are sent. The physical port is the only one connected to its agport. Non−PAgP packets are passed in and out between physical port and agport.
BiDir	Exactly one PAgP packet has been received that proves a bi−directional connection exists to exactly one neighbor. The physical port is not connected to any agport. PAgP packets are sent and may be received.
UpPAgP	This physical port, perhaps in association with other physical ports, is connected to an agport. PAgP packets are sent and received on the physical port. Non−PAgP packets are passed in and out between physical port and agport.

Both ends of both connections must agree on what the grouping is going to be, defined as the largest group of ports in the agport that is permitted by both ends of the connection.

When a physical port reaches the UpPAgP state, it is assigned to the agport that has member physical ports that match the new physical port's group−capability and that are in the BiDir or UpPAgP states. (Any such BiDir ports are moved to the UpPAgP state at the same time.) If there is no agport whose constituent physical port parameters are compatible with the newly ready physical port, it is assigned to an agport with suitable parameters that has no associated physical ports.

A PAgP timeout can occur on the last neighbor known on the physical port. The port timing out is removed from the agport. At the same time, all physical ports on the same agport whose timers have also timed out are removed. This enables an agport whose other end has died to be torn down all at once, instead of one physical port at a time.

Behavior in Failure

If a link in an existing channel is failed, (e.g. port unplugged, GBIC removed or fiber broken), then the agport is updated and the traffic is hashed over the remaining links without loss.

Note: The behavior when failing a link in a channel by powering off or removing a module may be different. By definition, there need to be two physical ports to a channel. If one port is lost from the system in a two−port channel, the logical agport is torn down and the original physical port is re−initialized with respect to spanning tree. This means traffic may be discarded until STP allows the port to become available to data again.

There is an exception to this rule on the Catalyst 6000. Before CatOS 6.3, an agport is not torn down during module removal if the channel is comprised of ports on modules 1 and 2 only.

This difference in the two failure modes is important when planning maintenance of a network, as there may be an STP Topology Change to consider when performing an on−line removal or insertion of a module. As stated in the previous section, it is important to manage each physical link in the channel with the NMS since the agport may remain undisturbed through a failure.

Some suggested steps to mitigate an unwanted topology change on the Catalyst 6000 are:

- If a single port is used per module to form a channel, three or more modules should be used (three ports or more total).
- If the channel spans two modules, two ports on each module should be used (four ports total).
- If a two−port channel is needed across two cards, use only the Supervisor ports.
- Upgrade to CatOS 6.3, which handles module removal without STP recalculation for channels split across modules.

Configuration Options

EtherChannels can be configured in different modes, summarized in the table below:

Mode	Configurable Options
On	PAgP not in operation. The port is channeling regardless of how the neighbor port is configured. If the neighbor port mode is on, a channel is formed.
Off	The port is not channeling regardless of how the neighbor is configured.
Auto (Default)	Aggregation is under control of the PAgP protocol. Places a port into a passive negotiating state, and no PAgP packets are sent on the interface until at least one PAgP packet is received that indicates that the sender is operating in desirable mode.
Desirable	Aggregation is under control of the PAgP protocol. Places a port into an active negotiating state, in which the port initiates negotiations with other ports by sending PAgP packets. A channel is formed with another port group in either desirable or auto mode.
Non−silent(Default on Catalyst 5000 fiber FE and GE ports)	An auto or desirable mode keyword. If no data packets are received on the interface, then the interface is never attached to an agport and cannot be used for data.This bi−directionality check was provided for specific Catalyst 5000 hardware as some link failures result in the channel being broken apart. By enabling non−silent mode, a recovering neighbor port is never allowed to come back up and break the channel apart unnecessarily.More flexible bundling and improved bi−directionality checks are present by default in Catalyst 4000 and 6000 series hardware. See section on auto−negotiation.

Silent(Default on all Catalyst 6000 and 4000 ports and 5000 copper ports)	An auto or desirable mode keyword. If no data packets are received on the interface, then after a 15 second timeout period, the interface is attached by itself to an agport and can thus be used for data transmission. Silent mode also allows for channel operation when the partner can be an analyzer or server that never sends PAgP.

The silent/non−silent settings affect how ports react to situations that cause unidirectional traffic or how they achieve a "nice fail−over." When a port is unable to transmit (because of a failed PHY or a broken fiber or cable, for example), this can still leave the neighbor port in an operational state. The partner continues to transmit data, but data is lost, as return traffic cannot be received. Spanning tree loops can also form because of the unidirectional nature of the link.

Some fiber ports have the desired capability of bringing the port to a non−operational state when it loses its receive signal (FEFI). This will cause the partner port to go non−operational and effectively causes the ports at both ends of the link to go down.

When using devices that will transmit data (such as BPDUs) and cannot detect unidirectional conditions, non−silent mode should be used to allow the ports to remain non−operational until receive data is present and the link is verified to be bi−directional. The time it takes for PAgP to detect a unidirectional link is around 3.5 * 30 seconds = 105 seconds, where 30 seconds is the time between two successive PAgP messages. UDLD is recommended as a more rapid detector to uni−directional links.

When using devices that will not transmit any data, silent mode should be used. This will force the port to become connected and operational regardless of whether received data is present or not. Additionally, for those ports that can detect the presence of a unidirectional condition, such as newer platforms using Layer 1 FEFI and UDLD, silent mode is used by default.

Verification

The following table depicts a summary of all the possible PAgP channeling mode scenarios between two directly connected switches (Switch−A and Switch−B). Some of these combinations may cause STP to put the ports on the channeling side into the ErrDisable state (that is, some of the combinations shut down the ports on the channeling side).

Switch−A Channel Mode	Switch−B Channel Mode	Channel State
On	On	Channel (non PAgP)
On	Off	Not Channel (ErrDisable)
On	Auto	Not Channel (ErrDisable)
On	Desirable	Not Channel (ErrDisable)
Off	On	Not Channel (ErrDisable)
Off	Off	Not Channel
Off	Auto	Not Channel

Off	Desirable	Not Channel
Auto	On	Not Channel (ErrDisable)
Auto	Off	Not Channel
Auto	Auto	Not Channel
Auto	Desirable	PAgP Channel
Desirable	On	Not Channel (ErrDisable)
Desirable	Off	Not Channel
Desirable	Auto	PAgP Channel
Desirable	Desirable	PAgP Channel

Recommendation

Cisco recommends enabling PAgP on all switch−to−switch channel connections, avoiding "on" mode. The preferred way is to set "desirable" mode at both ends of a link. The additional recommendation is to leave the "silent/non−silent" keyword at default − silent on Catalyst 6000s and 4000s, non−silent on Catalyst 5000 fiber ports.

As discussed in previous sections, explicitly configuring channeling off on all other ports is helpful for rapid data forwarding. Waiting up to 15 seconds for PAgP to timeout on a port that will not be used for channeling should be avoided, especially since the port is then handed over to STP, which itself can take 30 seconds to allow data forwarding, plus potentially 5 seconds for DTP for a total of 50 seconds. **Set port host** is discussed in more detail in the STP section.

```
set port channel <port range> mode desirable

set port channel <port range> mode off
!−−− Where ports not channeled, also part of the set port host command
```

This command assigns channels an **admin group** number, seen with a **show channel group** command. Addition and removal of channeling ports to the same agport can then be managed by referring to the admin number if desired.

Other Options

Another common configuration for customers who have a model of minimal administration at the access layer is to set the mode to "desirable" at the distribution and core layers and leave the access layer switches at the default "auto" configuration.

When channeling to devices that do not support PAgP, the channel needs to be hard−coded "on." This applies to devices such as servers, Local Director, Content Switches, routers, switches with older software, Catalyst XL switches, and Catalyst 8540s:

```
set port channel <port range> mode on
```

The new 802.3ad IEEE Link Aggregation Control Protocol (LACP) standard, available in CatOS 7.x, will likely supersede PAgP in the long term, as it brings the benefit of cross−platform and vendor interoperability.

Unidirectional Link Detection (UDLD)

The Uni−Directional Link Detection (UDLD) feature is intended to address several fault conditions on fiber and copper Ethernet interfaces:

- Monitoring physical cabling configurations and shutting down any mis−wired ports as *"ErrDisabled"*
- Protecting against uni−directional links. When a uni−directional link is detected, due to media or port/interface malfunction, the affected port is shut down as *"ErrDisabled"* and a corresponding syslog message generated.

See Understanding and Configuring the Unidirectional Link Detection Protocol (UDLD) Feature for more details.

Spanning tree, with its steady state unidirectional BPDU flow, was an acute sufferer from the above failures. It is easy to see how a port may suddenly be unable to transmit BPDUs, causing an STP state change from "blocking" to "forwarding" on the neighbor, which creates a loop, since the port is still able to receive.

Operational Overview

UDLD is a Layer 2 protocol working above the LLC layer (destination MAC 01−00−0c−cc−cc−cc, SNAP HDLC protocol type 0x0111). When running UDLD in combination with FEFI and auto−negotiation Layer 1 mechanisms, it is possible to validate the physical (Layer 1) and logical (Layer 2) integrity of a link.

UDLD has provisions for features and protection that FEFI and auto−negotiation cannot perform, namely the detection and caching of neighbor information, shutting down any misconnected ports, and detecting logical interface/port malfunctions or faults on links that are not point−to−point (those traversing media−converters or hubs).

UDLD employs two basic mechanisms: it learns about the neighbors and keeps the information up−to−date in a local cache, then sends a train of UDLD probe/echo (hello) messages whenever it detects a new neighbor or whenever a neighbor requests a re−synchronization of the cache.

UDLD constantly sends probe/echo messages on all ports. Whenever a corresponding UDLD message is received on a port, a "detection−phase" and validation process is triggered. If all valid conditions are met (the port is bi−directional and correctly wired), the port is enabled. If not, the port is error−disabled and a syslog message triggered similar to the following:

- "UDLD−3−DISABLE: Unidirectional link detected on port [dec]/[dec]. Port disabled"
- "UDLD−4−ONEWAYPATH: A unidirectional link from port [dec]/[dec] to port [dec]/[dec] of device [chars] was detected"

Once a link is established and classed as bi−directional, UDLD will continue to advertise probe/echo messages at a default interval of 15 seconds.

Port State	Comment
Undetermined	Detection in progress or neighboring UDLD has been disabled
Not applicable	UDLD has been disabled
Shutdown	Unidirectional link has been detected and the port disabled

Bi−directional	Bi−directional link has been detected

- **Neighbor Cache Maintenance:** UDLD periodically sends hello probe/echo packets on every active interface, in order to maintain the integrity of the UDLD neighbor cache. Whenever a hello message is received, it is cached and kept in memory for a maximum period defined as the hold−time. When the hold−time expires, the respective cache entry is aged out. If a new hello message is received within the hold−time period, the new one replaces the older entry and the corresponding time−to−live timer is reset.
- In order to maintain the integrity of the UDLD cache, whenever a UDLD−enabled interface gets disabled or a device is reset, all existing cache entries for the interfaces affected by the configuration change are cleared and UDLD transmits at least one message to inform respective neighbors to flush the corresponding cache entries.
- **Echo Detection Mechanism:** The echoing mechanism forms the basis of the detection algorithm. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out−of−synch neighbor, it starts/restarts the detection window on its side of the connection and sends a burst of echo messages in reply. Since this behavior must be the same across all neighbors, the echo sender expects to receive echos back in reply. If the detection window ends and no valid reply message has been received, the link is considered unidirectional, and a link re−establishment or port shutdown process may be triggered.

Convergence Time

To prevent spanning−tree loops, CatOS 5.4(3) reduced the UDLD default message interval from 60 seconds to 15 seconds in order to shut down a unidirectional link before a blocked port was able to transition to a forwarding state. The approximate time it takes for UDLD to detect a unidirectional failure is around (2.5 * message−interval + 4 seconds), or about 41 seconds using the default message interval of 15 seconds. This is well below the 50 seconds usually needed for STP to re−converge.

UDLD therefore has an assumed dependency on default spanning tree timers. If STP is tuned to converge more rapidly than UDLD, an alternate mechanism, like the CatOS 6.2 Loopguard feature, should be considered.

Note: Beware of older releases of UDLD that use a "non−configurable" 60−second default message interval, as they will thus be susceptible to spanning−tree loop conditions.

UDLD "aggressive−mode"

This feature provides enhanced protection against dangerous unidirectional link conditions in the following situations and includes attempts to re−establish a connection with the neighbor upon failure detection:

- One side of a link has a port stuck (either Tx and Rx).
- One side of a link remains up while the other side of the link has gone down. This reduces the reliance on Layer 1 FEFI mechanisms.
- After eight failed retries, the port is transitioned to an ErrDisable state and a syslog message logged.
- In these cases, UDLD aggressive mode will ErrDisable both of the ports on the link, which stops the loss of traffic.

Aggressive mode UDLD also allows the possibility of manually configuring the UDLD probe/echo message interval to values ranging from 7−90 seconds, the default interval being 15 seconds.

Recommendation

For maximum protection against symptoms resulting from uni−directional links, Cisco recommends enabling aggressive UDLD on point−to−point FE/GE links between Cisco switches, where the message interval is set to the 15−second default.

UDLD is disabled globally and enabled in readiness on fiber ports by default. As UDLD is an infrastructure protocol needed between switches only, it is disabled by default on copper ports, as these tend to be used for host access. Use the following commands to enable UDLD:

```
set udld enable
!−−− once globally enabled, all FE and GE fiber ports have UDLD enabled by default

set udld enable <port range>
!−−− for additional specific ports and copper media if needed

set udld aggressive−mode enable <port range>
!−−− all point to point links
```

Note: Switches that are not "aggressive−UDLD capable," currently the Cat2900−XL and Cat3500−XL, have hard−coded message−intervals of 60 seconds, which is not considered sufficiently fast to protect against potential spanning−tree loops (default STP parameters assumed).

UDLD is not easy to test without a genuinely faulty/unidirectional component in the lab, such as a defective GBIC. The protocol was designed to detect less common failure scenarios than those usually employed in a lab. For example, if performing a simple test such as unplugging one strand of a fiber in order to see the desired "ErrDisable" state, you should previously turn off Layer 1 auto−negotiation. Otherwise, the physical port will go down, thus resetting UDLD message communication, and the remote end will move to "undetermined" state rather than "ErrDisable."

Finally, if a port is placed in ErrDisabled state, by default it will remain down. It is worth noting the following command, which will re−enable ports after a time−out interval (300 seconds by default) if desired:

```
> (enable) set errdisable−timeout enable ?

bpdu−guard
!−−− BPDU Port−guard

channel−misconfig
!−−− Channel misconfiguration

duplex−mismatch

udld

other
!−−− Reasons other than the above

all
!−−− Apply ErrDisable timeout to all reasons
```

Ports that are error−disabled due to uni−directional link symptoms must be manually enabled using the **set port enable** command.

Other Options

If the partner device is not UDLD−capable, such as an end host or router, do not run the protocol:

```
set udld disable <port range>
```

Note: UDLD will not catch every STP failure situation, such as those caused by a CPU that does not send BPDUs for a time greater than (2 * FwdDelay + Maxage). Backup solutions like CatOS 6.2 Loopguard should be considered.

Furthermore, UDLD status and configuration consistency may be monitored using Cisco's UDLD SNMP MIB variables.

Management Configuration

Considerations to assist in controlling, provisioning, and troubleshooting a Catalyst network are discussed in this section.

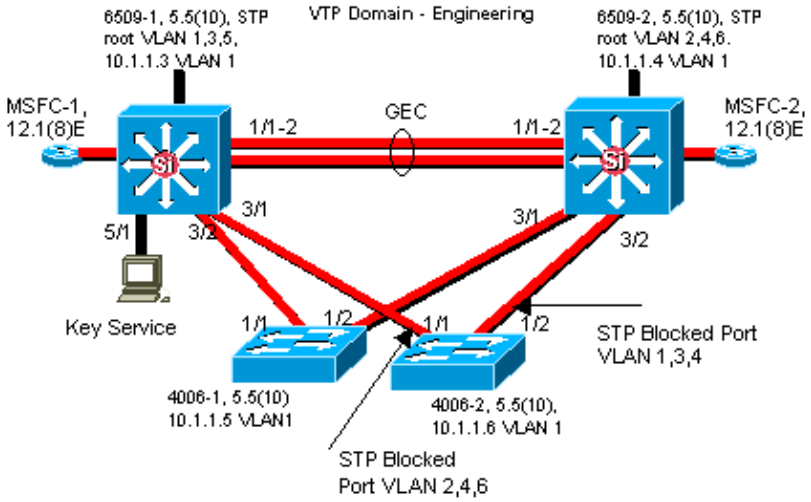
Network Diagrams

Clear network diagrams are a fundamental part of network operations. They become critical during troubleshooting and are the single most important vehicle for communicating information while escalating to vendors and partners during an outage. Their preparation, readiness, and accessibility should not be underestimated.

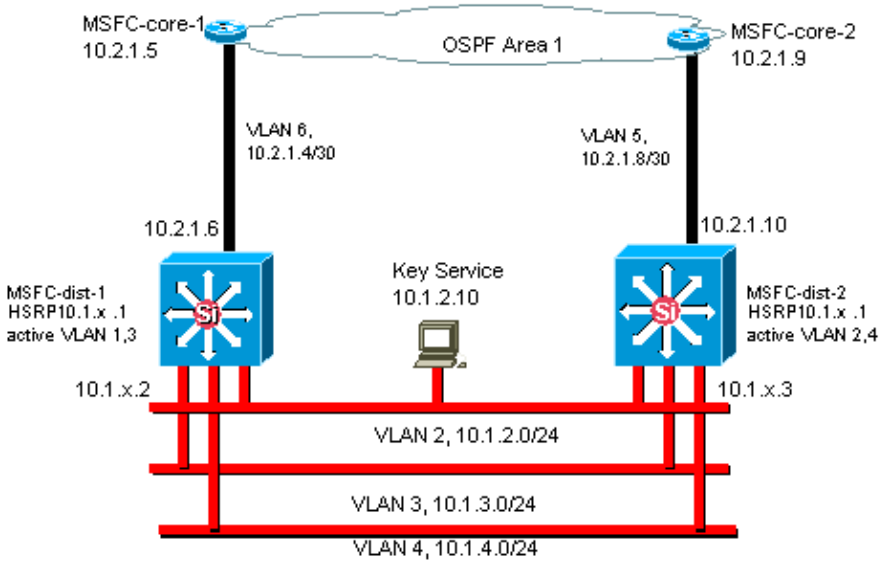
Recommendation

In Cisco experience, three types of diagram are needed:

- **Overall Diagram**—Even for the largest networks, a diagram that shows the end−to−end physical and logical connectivity is important. It can be common for enterprises that have implemented a hierarchical design to document each layer separately. However, during planning and problem solving, it is often a good knowledge of how the domains link together that matters.
- **Physical Diagram**—Shows all switch and router hardware and cabling. Trunks, links, speeds, channel groups, port numbers, slots, chassis types, software, VTP domains, root bridge, backup root bridge priority, MAC address, and blocked ports per VLAN should be labeled. It is often clearer to depict internal devices, such as the Catalyst 6000 Multilayer Switch Feature Card, as a "router on a stick" connected via a trunk.



- **Logical Diagram** − Shows only Layer 3 functionality (routers as objects, VLANs as Ethernet segments). IP addresses, subnets, secondary addressing, HSRP active and standby, access−core−distribution layers, and routing information should be labeled.



In−Band Management

Depending on the configuration, the switch in−band (internal) management interface (known as sc0) may have to handle the following data:

- Switch management protocols—SNMP, telnet, SSH, syslog
- User data—broadcasts and multicasts
- Switch control protocols—STP BPDUs, VTP, DTP, CDP, and so on.

It is common practice in Cisco's Multilayer design to configure a management VLAN that spans a switched domain and contains all "sc0" interfaces. This helps separate management traffic from user traffic and increases security of the switch management interfaces. This section describes the significance and potential problems of using the default VLAN 1 and running management traffic to the switch in the same VLAN as user traffic.

Operational Overview

The primary concern over the use of VLAN 1 for user data is that the Supervisor Network Management Processor (NMP) in general does not need to be interrupted by much of the multicast and broadcast traffic that is generated by end−stations. Older Catalyst 5000 hardware, the Supervisor I and Supervisor II in particular, has limited resources for dealing with this traffic, though the principle applies to all Supervisors. If the Supervisor CPU, buffer, or in−band channel to the backplane is fully occupied listening to unnecessary traffic, it is possible that control frames will be missed. In a worst−case scenario, this could lead to a spanning−tree loop or EtherChannel failure.

A **show interface** and **show ip stats** on the Catalyst can give some indication of the proportion of broadcast to unicast traffic and the proportion of IP to non−IP traffic (not typically seen in management VLANs).

A further health check for older Catalyst 5000 hardware is to examine the output of **show <inband | biga>** (hidden command) for resource errors (RscrcErrors), similar to buffer drops in a router. If these resource errors go up continuously, then memory is not available to receive system packets, perhaps because of a significant amount of broadcast traffic in the management VLAN. A single resource error may mean that the Supervisor is unable to process a packet such as Bridge Protocol Data Units (BPDUs), which could quickly become a problem, as protocols such at spanning−tree do not re−send missed BPDUs.

Recommendation

As highlighted in the opening chapter on VLAN 1, this special VLAN tags and handles most of the control plane traffic and is enabled on all trunks by default. With larger campus networks, care needs to be taken about the diameter of the VLAN 1 **STP domain**: instability in one part of the network could affect VLAN 1, thereby influencing control−plane stability and therefore STP stability for all other VLANs. Since CatOS 5.4, it has been possible to limit VLAN 1 from carrying user data and running STP using the following command:

```
clear trunk <mod/port> vlan 1
```

This does not stop control packets being sent from switch to switch in VLAN 1, as seen with a network analyzer. However, no data will be forwarded, and STP will not be run over this link. Therefore, this technique can be used to break VLAN 1 up into smaller failure domains.

Note: It is not currently possible to clear VLAN 1 trunks on 3500s and 2900−XLs.

Even if care has been taken with the campus design to constrain user VLANs to relatively small switch domains and correspondingly small failure/Layer 3 boundaries, some customers are still tempted to treat the management VLAN differently and try to cover the whole network with a single management subnet. There is no technical reason that a central NMS application must be Layer−2−adjacent to the devices it manages, nor is this a qualified security argument. Cisco recommends limiting the diameter of the management VLANs to the same routed domain structure as user VLANs and considering out−of−band management and/or CatOS 6.x SSH support as a way to increase network management security.

Other Options

There are design considerations for these Cisco recommendations in some topologies, however. For example, a desirable and common Cisco Multilayer design is one that avoids the use of an active spanning tree. This requires constraining each IP subnet/VLAN to a single access−layer switch, or cluster of switches. In these designs, there may be no trunking configured down to the access−layer.

There is no easy answer to the question of whether a separate management VLAN be created and trunking enabled to carry it between Layer 2 access and Layer 3 distribution layers. Here are two options for design

review with your Cisco engineer:

- **Option 1:** trunk two or three unique VLANs from the distribution layer down to each access−layer switch. This allows for a data VLAN, a voice VLAN, and a management VLAN, for example, and still has the benefit that STP is inactive. (Note that clearing VLAN 1 from trunks needs an extra configuration step.) In this solution, there are also design points to consider in order to avoid the temporary black−holing of routed traffic during failure recovery: STP PortFast for trunks (CatOS 7.x and onward) or VLAN Autostate synchronization with STP forwarding (after CatOS 5.5(9)).
- **Option 2:** having a single VLAN for data and management may be acceptable. With newer switch hardware, such as more powerful CPUs and control−plane rate−limiting controls, plus a design with relatively small broadcast domains as advocated by the MultiLayer design, the reality for many customers is that keeping the sc0 interface separate from the user data is less of an issue than it once was. A final decision is probably best taken by examining the broadcast traffic profile for that VLAN and discussing the capabilities of the switch hardware with your Cisco engineer**.** If the management VLAN does indeed contain all users on that access−layer switch**,** the use of IP input filters is highly recommended to secure the switch from users as per the security section.

Out−of−Band Management

Taking the arguments of the previous section one step further, network management can be made more highly available by constructing a separate management infrastructure around the production network so that devices are always reachable remotely no matter what traffic−driven or control−plane events occur. Two approaches are typical:

- Out−of−Band Management with an exclusive LAN
- Out−of−Band Management with Terminal Servers

Operational Overview

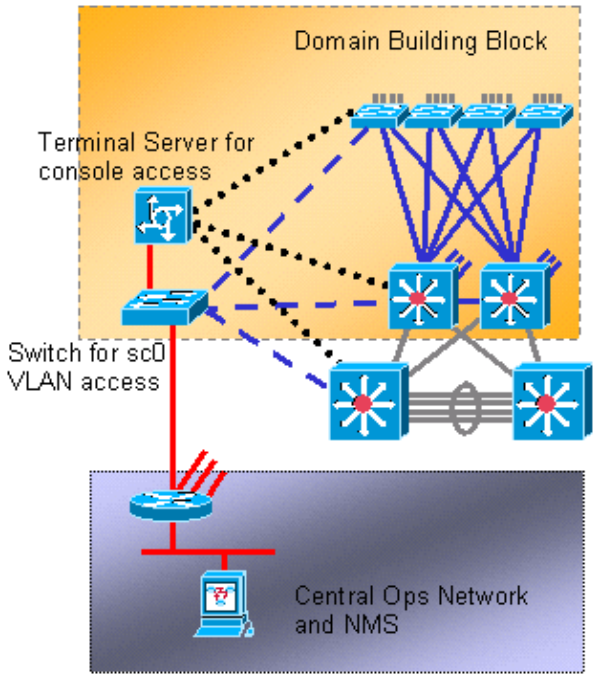
Every router and switch in the network can be provided with an out−of−band Ethernet management interface on a management VLAN. One Ethernet port on each device is configured in the management VLAN and cabled outside the production network to a separate switched management network via the sc0 interface. Note that Catalyst 4000 switches have a special "me1" interface on the Supervisor that is to be used for out−of−band management only, not as a switch port.

In addition, terminal server connectivity can be achieved by configuring a Cisco 2600 or 3600 with RJ−45−to−serial cables to access the console port of every router and switch in the layout. Using a terminal server also avoids the need for configuring backup scenarios, such as modems on auxiliary ports for every device. A single modem can be configured on the auxiliary port of the terminal server, thus providing dial−up service to the other devices during a network connectivity failure.

Recommendation

With this arrangement, two out−of−band paths to every switch and router will be possible in addition to numerous in−band paths, thus enabling highly−available network management. This architecture:

- Separates management traffic from user data
- Has the management IP address in a separate subnet, VLAN, and switch for higher security
- Provides higher assurance for management data delivery during network failures
- Has no active spanning tree in management VLAN: redundancy here is not critical



System Tests

Boot−up Diagnostics

During system boot−up, a number of processes are performed to ensure that a reliable and operational platform is available so that faulty hardware will not disrupt the network. Catalyst boot diagnostics are split between power−on self−test (POST) and online diagnostics.

Operational Overview

Depending on the platform and hardware configuration, different diagnostics are carried out at boot−up and when a card is "hot−swapped" into the chassis. A higher level of diagnostics will result in a wider number of problems detected but a longer boot cycle. Three levels of POST diagnostics may be selected (all tests check DRAM, RAM, and cache presence and size and initialize them):

Level	Additional Tests	Approximate Length (seconds)	Platform
Bypass	None	3	Not available on 4000 series using CatOS 5.5 or earlier
Minimal	Pattern−writing tests on the first MB of DRAM only	30	Default on 5000 and 6000 series; not available on 4000 series
Complete	Pattern−writing tests for all memory	60	Default on 4000 series

Online Diagnostics

These tests check packet paths internally in the switch. It is important to note that online diagnostics are therefore system−wide tests, not simply port tests. On Catalyst 5000s and 6000s, tests are performed first from the Standby Supervisor, then again from the Primary Supervisor. The length of the diagnostics depends on the system configuration (number of slots, modules, ports). There are three categories of test:

- Loopback test—Packets from the Supervisor NMP are sent to each port, then returned to the NMP and examined for errors.
- Bundling test—Channels of up to eight ports are created and loopback tests performed to the aggregate port to verify the hashing to specific links (see EtherChannel section for further information).
- Enhanced Address Recognition Logic (EARL) test—Both the central Supervisor and in−line Ethernet module Layer 3 rewrite engines are tested. Hardware forwarding entries and routed ports are created before sending sample packets (for each protocol encapsulation type) from the NMP via the switching hardware on each module and back to the NMP. This is for Catalyst 6000 PFC modules and newer.

"Complete" online diagnostics can take approximately 2 minutes. "Minimal" diagnostics do not perform bundle or re−write testing on modules other then the Supervisor, and can take approximately 90 seconds.

During a memory test, when a difference is found in the pattern read back compared to the pattern written, the port state is changed to "faulty." The results of these tests can be seen by issuing a **show test** command followed by the module number to be examined:

```
> show test 9

Diagnostic mode: complete (mode at next reset: complete)
!--- Configuration setting
Module 9 : 4-port Multilayer Switch
Line Card Status for Module 9 : PASS
Port Status :
  Ports 1  2  3  4
  ------------------
        .  .  .  .
Line Card Diag Status for Module 9 (. = Pass, F = Fail, N = N/A)
 Loopback Status [Reported by Module 1] :
  Ports 1  2  3  4
  ------------------
        .  . F  .
!--- Faulty  ^
 Channel Status :
  Ports 1  2  3  4
  ------------------
        .  .  .  .
```

Recommendation

Cisco recommends that all switches be set to use "complete" diagnostics to provide maximum fault detection and prevent outages during normal operations. **Note:** This change will not take effect until the next time the device is booted. Use the following command to set complete diagnostics:

```
set test diaglevel complete
```

Other Options

In some situations, a rapid boot−up time may be preferable over waiting to run full diagnostics. There are other factors and timings involved in bringing up a system, but overall, POST and online diagnostics add around a third again in time. In testing with a fully populated Single Supervisor nine−slot chassis with a Catalyst 6509, the total boot time was around 380 seconds with complete diagnostics, around 300 seconds with minimal diagnostics, and only 250 seconds with diagnostics bypassed. Bypass may be configured as follows:

```
set test diaglevel bypass
```

Note: The Catalyst 4000 will accept being configured for "minimal" diagnostics, though this still results in a complete test being undertaken. Minimal mode may be supported in the future on this platform.

Run time Diagnostics

Once the system is operational, the switch Supervisor performs various monitoring of the other modules. If a module is not reachable via the management messages (Serial Control Protocol (SCP) running over the out−of−band management bus), the Supervisor will attempt to restart the card or take other action as appropriate.

Operational Overview

The Supervisor carries out various monitoring automatically; this does not require any configuration. For the Catalyst 5000 and 6000, the following components of the switch are monitored:

- Network Management Processor via a watchdog
- Enhanced Addressed Resolution Logic (EARL) chip errors
- Inband channel from Supervisor to backplane
- Modules via keepalives over out−of−band channel (Catalyst 6000)
- Active Supervisor is monitored by the standby Supervisor for status (Catalyst 6000)
- **Note:** Since CatOS 6.2, further functionality has been added to this monitoring using the **set errordetection <inband | port | memory> enable** commands:
    - ♦ Each Supervisor sends SCP "pings" via the out−of−band channel to itself and the other Supervisor if present. Action is taken, such as resetting one of the Supervisors, if pings are constantly lost.
    - ♦ Each Supervisor sends ICMP "pings" via the in−band channel to itself and the other NMP and takes action if pings are constantly lost.

If any of these error conditions are seen, the switch writes a SYS or EARL type error message. For more information about error messages, refer to the Error Message Tool.

There is a similar function on the Catalyst 4000 platform, though its internal architecture is different.

Packet Buffer Diagnostics (Catalyst 5000 only)

This test is designed to find failed hardware on Catalyst 5000s that are using Ethernet modules with specific hardware that provide 10/100 Mbs connectivity between user ports and the switch backplane. As they cannot perform CRC checking for trunked frames, if a port packet buffer becomes defective during runtime, packets may get corrupted and cause CRC errors. Unfortunately, this could lead to the propagation of bad frames further into the Catalyst 5000 ISL network, potentially causing control plane disruption and broadcast storms in worst−case scenarios.

Newer Catalyst 5000 modules and other platforms have updated hardware error checking built in and do not need the packet buffer tests, so there is no option to configure it.

Line modules needing the Packet Buffer Diagnostics: WS−X5010, WS−X5011, WS−X5013, WS−X5020, WS−X5111, WS−X5113, WS−X5114, WS−X5201, WS−X5203, WS−X5213/a, WS−X5223, WS−X5224, WS−X5506, WS−X5509, WS−U5531, WS−U5533, WS−U5535

Operational Overview

This diagnostic checks that data stored in a specific section of the packet buffer is not accidentally being corrupted by faulty hardware. If the process reads back something different than it wrote, it shuts down the port in "Failed" mode, since that port may be corrupting data. There is no threshold of errors needed. Failed ports cannot be enabled again until the module has been reset (or replaced).

There are two modes for packet buffer tests: scheduled and on−demand. When a test begins, syslog messages are generated to indicate the expected length of the test (rounded up to the nearest minute) and the fact that the test has started. The exact length of the test varies by port type, size of the buffer, and the type of test run:

On−demand tests are aggressive in order to finish within a few minutes. Since these tests actively interfere with packet memory, ports must be administratively shut down before testing:

```
> (enable) test packetbuffer 4/1
Warning: only disabled ports may be tested on demand − 4/1 will be skipped.
> (enable) set port disable 4/1
> (enable) test packetbuffer 4/1
Packet buffer test started. Estimated test time: 1 minute.
%SYS−5−PKTTESTSTART:Packet buffer test started
%SYS−5−PKTTESTDONE:Packet buffer test done. Use 'show test' to see test results
```

Scheduled tests are much less aggressive than the on−demand tests, and they execute in the background. The tests are performed in parallel across multiple modules but on one port per module at a time. The test preserves, writes, and reads small sections of packet buffer memory before restoring user packet buffer data, and thus generates no errors. However, since the test is writing to buffer memory it will block incoming packets for a few milliseconds, thus causing some loss on busy links. By default there is an 8−second pause between each buffer−write test to minimize any packet loss, but this means that a system full of modules needing the packet buffer test may take over 24 hours for the test to complete. This scheduled test is enabled by default to run weekly at 03:30 on Sundays from CatOS 5.4 or later, and the test status can be confirmed with the following command:

```
>show test packetbuffer status

!−−− When test is running, command returns:
Current packet buffer test details
Test Type           : scheduled
Test Started        : 03:30:08 Jul 20 2001
Test Status         : 26% of ports tested
Ports under test    : 10/5,11/2
Estimated time left : 11 minutes
!−−− When test is not running, command returns:
Last packet buffer test details
Test Type           : scheduled
Test Started        : 03:30:08 Jul 20 2001
Test Finished       : 06:48:57 Jul 21 2001
```

Recommendation

The Cisco recommendation is to use the "scheduled packet buffer test" feature for Catalyst 5000 systems, as the benefit of discovering problems on modules outweighs the risk of low packet loss.

A standardized weekly time should then be scheduled across the network that will allow the customer to change links from faulty ports or RMA modules as necessary. As this test can cause some packet loss, depending on network load, it should be scheduled for quieter network times, such as 3:30 AM on a Sunday morning (which is the default). Use the following command to set the test time:

```
set test packetbuffer Sunday 3:30
!--- this is the default
```

Once enabled (as when upgrading to CatOS 5.4 and above for the first time), there is a chance that a previously hidden memory/hardware problem will be exposed and a port will be shut down automatically:

```
%SYS-3-PKTBUFBAD:Port 1/1 failed packet buffer test
```

Other Options

If it is not acceptable to risk a low level of packet loss per−port on a weekly basis, then it is recommended to use the on−demand feature during scheduled outages. Start this feature manually on a per range basis using the following command (though the port must be administratively disabled first):

```
test packetbuffer <port range>
```

System Logging

Syslog messages are Cisco−specific and a key part of proactive fault management. A wider range of network and protocol conditions are reported using syslog than is possible via standardized Simple Network Management Protocol (SNMP). Management platforms, such as Cisco Resource Manager Essentials and the Network Analysis Toolkit (NATkit), make powerful use of syslog information by:

- Presenting analysis by severity, message, device, etc.
- Enabling filtering of messages coming in for analysis
- Triggering alerting, such as pagers, or on−demand collecting of inventory and configuration changes

Recommendation

A particular point to focus on is what level of logging information is to be generated locally and held in the switch buffer as opposed to that which is sent to a syslog server (using the **set logging server severity <*value*>** command). Some organizations log a high level of information centrally, whereas others will go to the switch itself to look at the more detailed logs for an event or enable a higher level of syslog capture only during troubleshooting.

Debugging is different on CatOS platforms than IOS, but detailed system logging can be enabled on a per−session basis with **set logging session enable** without changing what is logged by default.

Cisco generally recommends bringing the "spantree" and "system" syslog facilities up to level 6, as these are key stability features to track. In addition, for multicast environments, bringing the logging level of the "mcast" facility up to 4 is recommended so that syslog messages are produced if router ports are deleted. Unfortunately, before CatOS 5.5(5) this could result in syslog messages being recorded for IGMP joins and leaves, which is too noisy to monitor. Finally, if IP input lists are used, a minimum logging level of 4 is

recommended to capture unauthorized login attempts. Use the following commands to set these options:

```
set logging buffer 500
!--- this is the default
set logging server <syslog server IP address>
set logging server enable
!--- this is the default
set logging timestamp enable
set logging level spantree 6 default
!--- increase default STP syslog level
set logging level sys 6 default
!--- increase default System syslog level
set logging server severity 4
!--- this is the default;
it will limit messages exported to syslog server
set logging console disable
```

Turning off console messages protects against the risk of the switch hanging as it waits for a response from a slow or non−existing terminal when message volume is high. Console logging is a high priority under CatOS and is mainly used to capture the final messages locally when troubleshooting or in a switch crash scenario.

Here are the individual logging facilities, default levels, and recommended changes for the Catalyst 6000. Each platform has slightly different facilities, depending on the features supported:

Facility	Default Level	Recommended Action
acl	5	Leave alone
cdp	4	Leave alone
cops	3	Leave alone
dtp	5	Leave alone
earl	2	Leave alone
filesys	2	Leave alone
gvrp	2	Leave alone
ip	2	**Change to 4 if IP input lists used**
kernel	2	Leave alone
ld	3	Leave alone
mcast	2	**Change to 4 if multicast used (CatOS 5.5(5) and onward)**
mgmt	5	Leave alone
mls	5	Leave alone
pagp	5	Leave alone
protfilt	2	Leave alone
pruning	2	Leave alone
Privatevlan	3	Leave alone
qos	3	Leave alone

radius	2	Leave alone
rsvp	3	Leave alone
security	2	Leave alone
snmp	2	Leave alone
spantree	2	**Change to 6**
sys	5	**Change to 6**
tac	2	Leave alone
tcp	2	Leave alone
telnet	2	Leave alone
Tftp	2	Leave alone
UDLD	4	Leave alone
VMPS	2	Leave alone
VTP	2	Leave alone

Note: Currently, the Catalyst switches log a configuration change syslog level−6 message for each **set** or **clear** command executed, unlike IOS, which triggers the message only after exiting configuration mode. If you need Resource Manager Essentials to back up configurations in real−time upon this trigger, then these messages also need to be sent to the RME syslog server. For most customers, however, periodic configuration backups for Catalyst switches are enough, and no change of the default server logging severity is needed.

If you are tuning your NMS alerts, you may wish to consult the System Message Guide. Also see Common CatOS Error Messages on Cisco Catalyst Switches and the Error Message Tool to help find error message explanations and recommendations.

Simple Network Management Protocol (SNMP)

SNMP is used to retrieve statistics, counters, and tables stored in network device Management Information Bases (MIBs). The information collected can be used by Network Management Stations (such as HP Openview) to generate real time alerts, measure availability, and produce capacity planning information, as well as helping to perform configuration and troubleshooting checks.

Operational Overview

With some security mechanisms, a network management station is able to retrieve information in the MIBs with SNMP protocol "get" and "get next" requests and to change parameters with a "set" command. Additionally, a network device can be configured to generate a trap message for the NMS for real time alerting. SNMP polling uses IP UDP port 161 and SNMP traps use port 162.

Cisco supports the following versions of SNMP:

- SNMPv1: RFC 1157 Internet Standard, using clear text community string security. An IP address access control list and password define the community of managers able to access the agent's MIB.
- SNMPv2C: a combination of SNMPv2, a draft Internet standard defined in RFCs 1902 through 1907, and SNMPv2C, a community−based administrative framework for SNMPv2 that is an experimental draft defined in RFC 1901. Benefits include a Bulk retrieval mechanism that supports the retrieval of tables and large quantities of information, minimizing the number of round−trips required, and improved error handling.

- SNMPv3: RFC 2570 proposed draft provides secure access to devices via the combination of authentication and encryption of packets over the network. The security features provided in SNMPv3 are:
    - ♦ Message integrity: ensuring that a packet has not been tampered with in−transit
    - ♦ Authentication: determining that the message is from a valid source
    - ♦ Encryption: scrambling the contents of a packet to prevent it from being viewed easily by an unauthorized source

The following table identifies the combinations of security models:

Model Level	Authentication	Encryption	Result
v1	noAuthNoPriv, Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv, Community String	No	Uses a community string match for authentication.
v3	noAuthNoPriv, Username	No	Uses a username match for authentication.
v3	authNoPriv, MD5 or SHA	No	Provides authentication based on the HMAC−MD5 or HMAC−SHA algorithms.
v3	authPriv, MD5 or SHA	DES	Provides authentication based on the HMAC−MD5 or HMAC−SHA algorithms. Provides DES 56−bit encryption in addition to authentication based on the CBC−DES (DES−56) standard.

Please note the following about SNMPv3 objects:

- Each user belongs to a group.
- A group defines the access policy for a set of users.
- An access policy defines what SNMP objects can be accessed for reading, writing, and creating.
- A group determines the list of notifications its users can receive.
- A group also defines the security model and security level for its users.

SNMP Trap Recommendation

SNMP is the foundation of all network management and is enabled and used on all networks. The SNMP agent on the switch must be set to use the version of SNMP supported by the management station. Since an agent can communicate with multiple managers, it is possible to configure the software to support communication with one management station using the SNMPv1 protocol and another using the SNMPv2 protocol, for example.

Most customers' Network Management Stations use SNMPv2C today under the following configuration:

```
set snmp community read-only <string>
!--- Allow viewing of variables only
```

```
set snmp community read-write <string>
!--- Allow setting of variables

set snmp community read-write-all <string>
!--- Include setting of SNMP strings
```

Cisco recommends enabling SNMP traps for all features in use (features not used can be disabled if desired). Once a trap is enabled, it can be tested using the **test snmp** command and appropriate handling set up on the NMS for the error (such as a pager alert or pop−up).

All traps are disabled by default and need to be added to the configuration, either individually or by using the **all** parameter:

```
set snmp trap enable all

set snmp trap <server address> <read-only community string>
```

Available traps in CatOS 5.5:

Trap	Description
auth	Authentication
bridge	Bridge
chassis	Chassis
config	Configuration
entity	Entity
ippermit	IP permit
module	Module
repeater	Repeater
stpx	Spanning Tree extension
syslog	Syslog notification
vmps	VLAN Membership Policy Server
vtp	VLAN Trunk Protocol

Note: The syslog trap will send all syslog messaged generated by the switch to the NMS as a SNMP trap also. If syslog alerting is already being performed by an analyzer such as Cisco Works 2000 Resource Manager Essentials, then it may not be useful to receive this information twice.

Unlike IOS, port level SNMP traps are disabled by default because switches can have hundreds of active interfaces. Cisco therefore recommends that key ports, such as infrastructure links to routers, switches, and main servers, have port−level SNMP traps enabled. Other ports, like user host ports, are not required, which helps to simplify network management.

```
set port trap <port range> enable
!--- enable on key ports only
```

SNMP Polling Recommendation

A network management review is recommended to discuss specific needs in detail, but some basic Cisco philosophies in managing large networks are:

- Do something simple, and do it well.
- Reduce staff overload due to excessive data polling, collection, tools, and manual analysis.
- Network management is possible with just a few tools: HP Openview as an NMS; Cisco Resource Manager Essentials as a configuration, syslog, inventory, and software manager; Microsoft Excel as an NMS data analyzer; CGI as a way to publish to the web.
- Publishing reports to the web allows "users," such as senior management and analysts, to help themselves to information without burdening operations staff with many special requests.
- Find out what is working well on the network and leave it alone. Concentrate on what is not working.

The first phase of implementing NMS must be to baseline the network hardware. Much can be inferred about device and protocol health from simple CPU, memory, and buffer utilization on routers, and NMP CPU, memory, and backplane utilization on switches. Only after a hardware baseline do Layer 2 and Layer 3 traffic load, peak, and average baselines become fully meaningful. Baselines are usually established over several months to get visibility of daily, weekly, and quarterly trends – according to the business cycle of the company.

Many networks suffer NMS performance and capacity problems caused by over−polling. It is therefore recommended, once the baseline is established, to set alarm and event RMON thresholds on the devices themselves to alert the NMS on abnormal changes, thus removing polling. This enables the network to tell the operators when something is not normal rather than continuously polling to see whether everything is normal. Thresholds can be set based on various rules, such as maximum value plus a percentage or standard deviation from a mean, and are outside the scope of this document.

The second phase of implementing NMS is to poll particular areas of the network in more detail with SNMP. This includes areas of doubt, areas before a change, or areas that should be characterized as working well. Use the NMS systems as a "searchlight" to scan the network in detail and illuminate hot spots (do not attempt to light up the whole network).

The Cisco Network Management Consulting group has suggested the following key fault MIBs to be analyzed or monitored in campus networks. More information (on performance MIBs to poll, for example) is available at Cisco Network Monitoring and Event Correlation Guidelines.

Object Name	Object Description	OID	Poll Interval	Threshold
MIB−II				
sysUpTime	system uptime in 1/100ths of seconds	1.3.6.1.2.1.1.3	5 min	< 30000
CISCO−PROCESS−MIB				
cpmCPUTotal5min	The overall CPU busy percentage in the last 5 minute period	1.3.6.1.4.1.9.9.109.1.1.1.1.5	10 min	Baseline
CISCO−STACK−MIB				
sysEnableChassisTraps	Indicates whether chassisAlarmOn and chassisAlarmOff traps in this MIB should be generated.	1.3.6.1.4.1.9.5.1.1.24	24 hrs	1
sysEnableModuleTraps	Indicates whether moduleUp and moduleDown traps in this MIB should be generated.	1.3.6.1.4.1.9.5.1.1.25	24 hrs	1

sysEnableBridgeTraps	Indicates whether newRoot and topologyChange traps in the BRIDGE−MIB (RFC 1493) should be generated.	1.3.6.1.4.1.9.5.1.1.26	24 hrs	1
sysEnableRepeaterTraps	Indicates whether the traps in the REPEATER−MIB (RFC1516) should be generated.	1.3.6.1.4.1.9.5.1.1.29	24 hrs	1
sysEnableIpPermitTraps	Indicates whether the IP permit traps in this MIB should be generated.	1.3.6.1.4.1.9.5.1.1.31	24 hrs	1
sysEnableVmpsTraps	Indicates whether the vmVmpsChange trap defined in CISCO−VLAN−MEMBERSHIP−MIB should be generated.	1.3.6.1.4.1.9.5.1.1.33	24 hrs	1
sysEnableConfigTraps	Indicates whether sysConfigChange trap in this MIB should be generated.	1.3.6.1.4.1.9.5.1.1.35	24 hrs	1
sysEnableStpxTrap	Indicates whether stpxInconsistencyUpdate trap in the CISCO−STP−EXTENSIONS−MIB should be generated.	1.3.6.1.4.1.9.5.1.1.40	24 hrs	1
chassisPs1status	Status of power supply 1	1.3.6.1.4.1.9.5.1.2.4	10 min	2
chassisPs1TestResult	Detailed information on status of power supply 1	1.3.6.1.4.1.9.5.1.2.5	As needed	
chassisPs2Status	Status of power supply 2	1.3.6.1.4.1.9.5.1.2.7	10 min	2
chassisPs2TestResult	Detailed information on status of power supply 2	1.3.6.1.4.1.9.5.1.2.8	As needed	
chassisFanStatus	Status of Chassis Fan	1.3.6.1.4.1.9.5.1.2.9	10 min	2
chassisFanTestResult	Detailed information on status of chassis fan	1.3.6.1.4.1.9.5.1.2.10	As needed	
chassisMinorAlarm	Chassis Minor Alarm Status	1.3.6.1.4.1.9.5.1.2.11	10 min	1
chassis MajorAlarm	Chassis Major Alarm Status	1.3.6.1.4.1.9.5.1.2.12	10 min	1
chassisTempAlarm	Chassis Temperature Alarm status	1.3.6.1.4.1.9.5.1.2.13	10 min	1
moduleStatus	Operational Status of the module	1.3.6.1.4.1.9.5.1.3.1.1.10	30 min	2
moduleTestResult	Detailed information on modules condition	1.3.6.1.4.1.9.5.7.3.1.1.11	As needed	
moduleStandbyStatus	Status of a redundant module	1.3.6.1.4.1.9.5.7.3.1.1.21	30 min	=1 or =4
BRIDGE−MIB				
dot1dStpTimeSinceTopologyChange	The time (in 1/100 secs) since the last time a topology change was detected by the entity	1.3.6.1.2.1.17.2.3	5 min	< 30000
dot1dStpTopChanges	The total number of topology changes detected by this bridge since the management entity was last reset or initialized	1.3.6.1.2.1.17.2.4	As needed	
dot1dStpPortState [1]	Port's current state as defined by application of the Spanning Tree Protocol. Return value can be one of the following: disabled (1), blocking (2), listening (3), learning (4), forwarding (5), or broken (6).	1.3.6.1.2.1.17.2.15.1.3	As needed	

CISCO−MEMORY−POOL−MIB				
ciscoMemoryPoolUsed	Indicates the number of bytes from the memory pool that are currently in use by applications on the managed device	1.3.6.1.4.1.9.9.48.1.1.1.5	30 min	Baseline
ciscoMemoryPoolFree	Indicates the number of bytes from the memory pool that are currently unused on the managed device. Note that the sum of ciscoMemoryPoolUsed and ciscoMemoryPoolFree is the total amount of memory in the pool	1.3.6.1.4.1.9.9.48.1.1.1.6	30 min	Baseline
ciscoMemoryPoolLargestFree	Indicates the largest number of contiguous bytes from the memory pool that are currently unused on the managed device	1.3.6.1.4.1.9.9.48.1.1.1.7	30 min	Baseline

For more information on Cisco MIB support, see the Cisco Network Management Toolkit − MIBs.

Note: Some standard MIBs assume that a particular SNMP entity contains only one instance of the MIB. Thus, the standard MIB does not have any index that would allow users to directly access a particular instance of the MIB. In these cases, community string indexing is provided to access each instance of the standard MIB. The syntax is [community string]@[instance number], where instance is typically a VLAN number.

Other Options

The security aspects of SNMPv3 mean that its use is expected to overtake SNMPv2 in time, so Cisco recommends that customers prepare for this new protocol as part of their NMS strategy. The benefits are that data can be collected securely from SNMP devices without fear of tampering or corruption. Confidential information, such as SNMP "**set** command" packets that change a switch configuration, can be encrypted to prevent its contents from being exposed on the network. In addition, different user groups can have different privileges.

Note: The configuration of SNMPv3 is significantly different than the SNMPv2 command line, and increased CPU load on the Supervisor is to be expected.

Remote Monitoring (RMON)

RMON permits the pre−processing of MIB data by the network device itself, in preparation for common uses or application of that information by the network manager, such as performing historical baseline determination and threshold analysis.

The results of RMON processing are stored in RMON MIBs for subsequent collection by an NMS, as defined in RFC 1757.

Operational Overview

Catalyst switches support "mini−RMON" in hardware on each port, which consists of four basic RMON−1 groups: Statistics (group 1), History (group 2), Alarms (group 3), and Events (group 9).

The most powerful part of RMON−1 is the **threshold mechanism** provided by the **alarm and event** groups. As discussed in the previous section, the configuration of RMON thresholds allows the switch to send an SNMP trap when an anomalous condition occurs. Once key ports have been identified, SNMP can be used to poll counters or RMON history groups and create baselines recording normal traffic activity for those ports. Next, RMON rising and falling thresholds can be set and alarms configured for when there is a defined variance from the baseline.

Configuration of thresholds is best done using an RMON management package, since successfully creating the rows of parameters in Alarm and Event tables is tedious. Commercial RMON NMS packages, such as Cisco's Traffic Director, part of Cisco Works 2000, incorporate graphical user interfaces (GUIs) that make the setting of RMON thresholds much simpler.

For baseline purposes, the "etherStats" group provides a useful range of Layer 2 traffic statistics. The objects in this table can be used to get statistics on unicast, multicast, and broadcast traffic as well as a variety of Layer 2 errors. The RMON agent on the switch can also be configured to store these sampled values in the history group. This mechanism enables the amount of polling to be reduced without reducing the sample rate. Using RMON histories can give accurate baselines without substantial polling overhead. However, the more histories collected, the more switch resources are used.

While switches provide only four basic groups of RMON−1, it is important not to forget the rest of RMON−1 and RMON−2. All groups are defined in RFC 2021, including UsrHistory (group 18) and ProbeConfig (group 19)). Layer 3 and higher information can be retrieved from switches using the SPAN port or VLAN ACL redirect features to copy traffic to an external RMON SwitchProbe or an internal Network Analysis Module (NAM).

NAMs support all RMON groups and can even examine **application layer data**, including Netflow data exported from Catalysts when Multi−Layer Switching is enabled. Running MLS means that the router will not switch all packets in a flow, so only Netflow data−export and not interface counters will give reliable VLAN accounting.

You can use a SPAN port and a Switch Probe to capture a packet stream for a particular port, trunk, or VLAN and upload the packets for decoding by an RMON management package. The SPAN port is SNMP−controllable via the SPAN group in the CISCO−STACK−MIB, so this process is easy to automate. Traffic Director makes use of these features with its "roving agent" feature. For more details, see Network Monitoring in an Enterprise LAN Environment.

There are caveats to spanning a whole VLAN. Even if you use a 1Gbps probe, the entire packet stream from one VLAN or even one 1Gbps full−duplex port may exceed the bandwidth of the SPAN port. If the SPAN port is continuously running at full bandwidth, chances are data is being lost. For more details, see Configuring the Catalyst Switched Port Analyzer (SPAN) Feature.

Recommendation

Cisco recommends that RMON thresholds and alerting be deployed to help network management in a more intelligent way than SNMP polling alone. This will reduce network management traffic overhead and allow the network to alert intelligently when something has changed from the baseline. RMON needs to be driven by an external agent such as Traffic Director − there is no command line interface support. Use the following

commands to enable RMON:

```
set snmp rmon enable
set snmp extendedrmon netflow enable <mod>
!--- for use with NAM module only
```

It is important to remember that the primary function of a switch is to forward frames, not to act as a large multi−port RMON probe. Therefore, as you are setting up histories and thresholds on multiple ports for multiple conditions, keep in mind that resources are being consumed. Consider a NAM module if you are scaling up RMON. Also remember the critical port rule: only poll and set thresholds on the ports identified as important in the planning stage.

Memory requirements

RMON memory usage is constant across all switch platforms relating to statistics, histories, alarms, and events. RMON uses a "bucket" to store histories and statistics on the RMON agent (the switch, in this case). The bucket size is defined on the RMON probe (Switch Probe) or RMON application (Traffic Director), then sent to the switch to be set. Typically, memory constraints are only a consideration on older Supervisors with less than 32MB of DRAM. Here are further guidelines:

- Approximately 450K of code space is added to the NMP image in order to support mini−RMON (which is four groups of RMON: statistics, history, alarms, and events). The dynamic memory requirement for RMON varies because it depends on the run−time configuration. The run−time RMON memory usage information for each mini−RMON group is explained below:
  - ♦ Ethernet Statistics group—Takes 800 bytes for each switched Ethernet/Fast Ethernet interface.
  - ♦ History group—For the Ethernet interface, each configured history control entry with 50 buckets takes approximately 3.6KB memory space and 56 bytes for each additional bucket.
  - ♦ Alarms and Events groups—Takes 2.6KB for each configured alarm and its corresponding event entries.
- Saving the RMON−related configuration takes approximately 20K NVRAM of space if the system total NVRAM size is 256K or more and 10K NVRAM of space if the total NVRAM size is 128K.

Network Time Protocol (NTP)

The Network Time Protocol, RFC 1305, synchronizes timekeeping among a set of distributed time−servers and clients and allows events to be correlated when system logs are created or other time−specific events occur.

NTP provides client time accuracies, typically within a millisecond on LANs and up to a few tens of milliseconds on WANs, relative to a primary server synchronized to Coordinated Universal Time (UTC). Typical NTP configurations utilize multiple redundant servers and diverse network paths in order to achieve high accuracy and reliability. Some configurations include cryptographic authentication to prevent accidental or malicious protocol attacks.

Operational Overview

The Network Time Protocol was first documented in RFC−958, but has evolved through RFC−1119 (NTP version 2) and is now in its third version as defined in RFC−1305. It runs over the User Datagram Protocol (UDP port 123). All NTP communication uses Coordinated Universal Time (UTC), which is the same time as Greenwich Mean Time.

Accessing Public Time Servers

The NTP subnet presently includes over 50 public primary servers synchronized directly to UTC by radio, satellite, or modem. Normally, client workstations and servers with a relatively small number of clients do not synchronize to primary servers. There are about 100 public secondary servers synchronized to the primary servers that provide synchronization to over 100,000 clients and servers on the Internet. The current lists are maintained on the List of Public NTP Servers page, which is updated regularly. There are numerous private primary and secondary servers not normally available to the public as well. For a list of public NTP servers and information about using them, consult the University of Delaware's Time Synchronization Server page .

Since there is no guarantee that these public Internet NTP servers will be available, or that they will produce the correct time, it is strongly advised that other options be considered. This could include making use of various standalone Global Positioning Service (GPS) devices directly connected to a number of routers.

Another possible option is the use of various routers configured as Stratum 1 masters, although this is not recommended.

Stratum

Each NTP server adopts a "stratum" that indicates how far away from an external source of time the server is. Stratum 1 servers have access to some kind of external time source, such as a radio clock. Stratum 2 servers obtain time details from a nominated set of Stratum 1 servers, while Stratum 3 servers obtain time details from Stratum 2 servers, and so on.

Server Peer Relationship

- A "server" is one that will respond to client requests, but will not try to incorporate any date information from a client time source.
- A "peer" is one that will respond to client requests, but will try to use the client requests as being a potential candidate for a better time source and to aid in stabilization of its clock frequency.
- To be a true peer, both sides of the connection should enter into a peer relationship rather than have one user a peer and the other user a server. It is also recommended that peers exchange keys so that only trusted hosts will talk to each other as peers.
- In a client request to a server, the server will answer the client and forget that the client ever asked a question; in a client request to a peer, the server will answer the client and keep state information about the client to track how well it is doing at timekeeping and what stratum server it is running. **Note:** CatOS can only act as an NTP client.

It is no problem for an NTP server to handle many thousands of clients. However, handling hundreds of peers will have memory impact, and the state maintenance will consume more CPU resources on the box as well as bandwidth.

Polling

The NTP protocol allows a client to query a server any time it wishes. In fact, when NTP is first configured in a Cisco device, it sends out eight queries in rapid succession at NTP_MINPOLL (2^4 = 16 second) intervals. The NTP_MAXPOLL is 2^{14} seconds (which is 16384 seconds or 4 hours 33 minutes 4 seconds), the maximum time it will take before NTP will poll again for a response. At present, Cisco does not have a method of manually forcing the POLL time to be set by the user.

The NTP polling counter starts at 2^6 (64) seconds and is incremented by powers of two (as the two servers sync with each other), to 2^{10}. That is, you may expect the sync messages to be sent at an interval of 64, 128,

256, 512, or 1024 seconds per configured server or peer. The time varies between 64 seconds and 1024 seconds as a power of two based on the phase−locked−loop that sends and receives packets. If there is a lot of jitter in the time, it will poll more often. If the reference clock is accurate and the network connectivity consistent, you should see the poll−times converge on 1024 seconds between each poll.

In the real world, this means that the NTP Poll Interval changes as the connection between the client and server changes. The better the connection, the longer the poll interval, meaning that the NTP client has received 8 responses for its last 8 requests (the poll interval will then be doubled). A single missed response will cause the poll interval to be halved. The poll interval starts out at 64 seconds and goes to a maximum of 1024 seconds. In the best circumstances, it will take a little over 2 hours for the poll interval to go from 64 seconds to 1024 seconds.

Broadcasts

NTP broadcasts are *never* forwarded. The **ntp broadcast** command will cause the router to originate NTP broadcasts on the interface on which it is configured. The **ntp broadcast client** command will cause the router or switch to listen to NTP broadcasts on the interface on which it is configured.

NTP Traffic Levels

The bandwidth utilized by NTP is minimal, since the interval between polling messages exchanged between peers usually ratchets back to no more than one message every 17 minutes (1024 seconds). With careful planning, this can be maintained within router networks over the WAN links. The NTP clients should peer to local NTP servers, not all the way across the WAN to the central site core routers who will be the stratum 2 servers.

A converged NTP client will use approximately 0.6 bits/second per server.

Recommendation

Many customers have NTP configured in client mode today on their CatOS platforms, synchronized from several reliable feeds from the Internet or a radio clock. However, a simpler alternative to server mode when operating a large number of switches is to enable NTP in broadcast client mode on the management VLAN in a switched domain. This mechanism allows an entire domain of Catalysts to receive a clock from a single broadcast message; however, the accuracy of timekeeping is marginally reduced because the information flow is one way.

Using loopback addresses as the source of updates can also help with consistency. Security concerns can be addressed in the following two ways:

- Filtering server updates
- Authentication

Time correlation of events is extremely valuable in two cases: troubleshooting and security audits. Care should be taken to protect the time sources and data, and encryption is recommended so that key events are not erased either intentionally or unintentionally.

Cisco recommends the following configurations:

Catalyst configuration
`set ntp broadcastclient enable` `set ntp authentication enable`

```
set ntp key <key>
!--- This is an MD5 hash
set ntp timezone <zone name>
set ntp summertime <date change details>
```

Alternate Catalyst configuration

```
!--- This more traditional configuration creates
more configuration work and NTP peerings
set ntp client enable
set ntp server <IP address of time server>
set timezone <zone name>
set summertime <date change details>
```

Router configuration

```
!--- This is a sample router configuration to distribute
NTP broadcast information to the Catalyst broadcast clients
ntp source loopback0
ntp server <IP address of time server>
ntp update-calendar
clock timezone <zone name>
clock summer-time <date change details>
ntp authentication key <key>
ntp access-group <access-list>
!--- to filter updates to allow only trusted sources of NTP information
Interface <to campus/management VLAN containing switch sc0>
       ntp broadcast
```

Cisco Discovery Protocol (CDP)

CDP exchanges information between adjacent devices over the data link layer and is extremely helpful in determining network topology and physical configuration outside of the logical or IP layer. Supported devices are mainly switches, routers, and IP phones. This section highlights some of the enhancements of CDP version 2 over version 1.

Operational Overview

CDP uses SNAP encapsulation with type code 2000. On Ethernet, ATM, and FDDI, the destination multicast address **01−00−0c−cc−cc−cc, HDLC protocol type 0x2000** is used. On token rings, the functional address c000.0800.0000 is used. CDP frames are sent periodically every minute by default.

CDP messages contain one or more sub−messages that allow the destination devices to gather and store information about every neighbor device.

CDP version 1 supports the following parameters:

Parameter	Type	Description
1	Device−ID	Hostname of the device or hardware serial number in ASCII
2	Address	The Layer 3 address of the interface that has sent the update

3	Port−ID	The port on which the CDP update has been sent
4	Capabilities	Describes the device's functional capabilities: Router: 0x01 TB Bridge: 0x02 SR Bridge: 0x04 Switch: 0x08 (Provides Layer 2 and/or Layer 3 switching) Host: 0x10 IGMP conditional filtering: 0x20 The Bridge or Switch does not forward IGMP report packets on non−routerports. Repeater: 0x40
5	Version	A character string containing the software version (same as in **show version**)
6	Platform	Hardware platform, such as WS−C5000, WS−C6009, or Cisco RSP

In CDP version 2, additional protocol fields have been introduced. CDP version 2 supports any field, but the following can be particularly useful in switched environments and are used in CatOS. Note that when a switch runs CDPv1, it will drop v2 frames. When a switch running CDPv2 receives a CDPv1 frame on an interface, it will start sending out CDPv1 frames out of that interface in addition to CDPv2 frames.

Parameter	Type	Description
9	VTP Domain	The VTP Domain, if configured on the device.
10	Native VLAN	In dot1Q, this is the untagged VLAN.
11	Full/Half Duplex	This field contains the duplex setting of the sending port.

Recommendation

CDP is enabled by default and is essential to gain visibility of adjacent devices and for troubleshooting. It is also used by network management applications to build Layer 2 topology maps. Use the following commands to set up CDP:

```
set cdp enable
!−−− this is the default
set cdp version v2
!−−− this is the default
```

In parts of the network where a high level of security is required (such as Internet−facing DMZs), CDP should be turned off as follows:

```
set cdp disable <port range>
```

The command **show CDP neighbors** displays the local CDP table. Entries marked with a star (*) indicate a VLAN mismatch; entries marked with a # indicate a duplex mismatch. This can be a valuable help for troubleshooting.

```
> show cdp neighbors

* - indicates vlan mismatch.
# - indicates duplex mismatch.
Port  Device-ID          Port-ID Platform
----- ------------------ ------- ------------
 3/1  TBA04060103(swi-2) 3/1     WS-C6506
 3/8  TBA03300081(swi-3) 1/1     WS-C6506
15/1  rtr-1-msfc         VLAN 1  cisco    Cat6k-MSFC
16/1  MSFC1b             Vlan2   cisco    Cat6k-MSFC
```

Other Options

Some switches, like the Catalyst 6500, have the ability to supply power via UTP cables to IP Phones. Information received via CDP assists power management on the switch.

As IP Phones might have a PC connected to them, and both devices connect to the same port on the Catalyst, the switch has the ability to put the VoIP phone in a separate VLAN, the "auxiliary." This allows the switch to easily apply a different Quality of Service for the VoIP traffic.

In addition, if the auxiliary VLAN is modified (for example, to force the phone to use a specific VLAN or specific tagging method), this information is sent to the phone via CDP.

Parameter	Type	Description
14	Appliance ID	Allows the VoIP traffic to be differentiated from other traffic, as by separate VLAN-id (auxiliary VLAN).
16	Power Consumption	The amount of power a VoIP phone consumes, in milliwatts

Note: Catalyst 2900 and 3500 XL switches do not currently support CDPv2.

Security Configuration

Ideally, the customer should have already established a Security Policy to help define what tools and technologies from Cisco are qualified.

Note that IOS security, as opposed to CatOS, is dealt with in many papers, such as Cisco ISP Essentials.

Basic Security Features

Passwords

First, configure a user level password (login). Passwords are case sensitive from CatOS 5.x onward and may be from 0 to 30 characters in length, including spaces. Next, set the enable password:

```
set password <password>

set enablepass <password>
```

All passwords should meet minimum length standards (for example: 6 characters minimum, a mix of letters and numbers, upper- and lower-case letters) for login and enable passwords when used.

In order to allow for more flexibility in managing password security and device access, Cisco recommends the use of a TACACS+ Server. Please refer to the TACACS+ section of this document for more information.

IP Permit Filters

These are filters to safeguard access to the management sc0 interface via telnet and other protocols. These are particularly important when the VLAN used for management also contains users. Use the following commands to enable IP address and port filtering:

```
set ip permit enable

set ip permit <IP address> <mask> <telnet|ssh|snmp|all>
```

However, restricting telnet access with the above command means that access to CatOS devices may only be via a few trusted end−stations, which may be a hindrance in troubleshooting. Please keep in mind that it is possible to spoof IP addresses and to fool filtered access, so this is the first layer of protection only.

It is recommended to log unauthorized access using the syslog messages, described earlier, by setting the "ip" facility logging level to 4 from its default of 2.

Port Security

Consider utilizing port security to permit only one or several known MAC addresses to pass data on a particular port (to stop static end stations from being swapped for new stations without change control, for example). This is possible either using static MAC addresses:

```
set port security <mod/port> enable <MAC address>
```

or by learning restricted MAC addresses dynamically:

```
set port security <port range> enable
```

The following options can be configured:

- **set port security** <***mod/port***> **age** <***time value***>**—**Specifies the duration for which addresses on the port will be secured before a new address can be learned. Valid time in minutes is 10 − 1440. Default is no aging.
- **set port security** <***mod/port***> **maximum** <***value***>**—**Keyword that specifies the maximum number of MAC addresses to secure on the port. Valid values are 1 (default) − 1025.
- **set port security** <***mod/port***> **violation shutdown—**Shuts down port (default) if violation occurs as well as sending syslog message (default) and discarding the traffic.
- **set port security** <***mod/port***> **shutdown** <***time value***>**—**Duration for which a port will remain disabled. Valid values are 10 − 1440 minutes. Default is permanently shutdown

Since CatOS 6.x, Cisco has introduced 802.1x authentication that allows clients to authenticate to a central server before ports can be enabled for data. This feature is in the early stages of support on such platforms as Windows XP, but may be considered a strategic direction by many enterprises.

Login Banners

Create appropriate device banners to state specifically the actions taken for unauthorized access. Do not advertise site name or network data that may provide information to unauthorized users. These banners provide recourse in case a device is compromised and the perpetrator is caught:

```
# set banner motd ^C
*** Unauthorized Access Prohibited ***
***  All transactions are logged   ***
------------- Notice Board -------------
----Contact Joe Cisco at 1 800 go cisco for access problems----
^C
```

Physical Security

Devices should not be accessible physically without proper authorization, so the equipment should be in a controlled (locked) space. To ensure that the network stays operational and unaffected by malicious tampering of environmental factors, all equipment should have proper UPS (Uninterruptible Power Supply with redundant sources where possible) and temperature control (air conditioning). Remember, if physical access is breached by a person with malicious intent, disruption via password recovery or other methods is much more likely.

Terminal Access Controller Access Control System (TACACS+)

By default, non−privileged and privileged mode passwords are global and apply to every user accessing the switch or router, either from the console port or via a telnet session across the network. Their implementation on network devices is time consuming and non−centralized. It is also difficult to implement access restrictions using access lists that can be prone to configuration errors.

Three security systems are available to help control and police access to network devices. These use client/server architectures to place all security information in a single central database:

- Terminal Access Controller Access Control System (TACACS+)
- Remote Authentication Dial−in User Service (RADIUS)
- Kerberos

TACACS+ is a common deployment in Cisco networks and is the focus of this chapter. It provides the following features:

- Authentication—The process of identifying and verifying a user. Several methods can be used to authenticate a user, but the most common includes a combination of user name and password.
- Authorization—Of various commands can be granted once a user is authenticated.
- Accounting—The recording what a user is doing or has done on the device.

See Configuring TACACS+, RADIUS, and Kerberos on Catalyst Switches for more details.

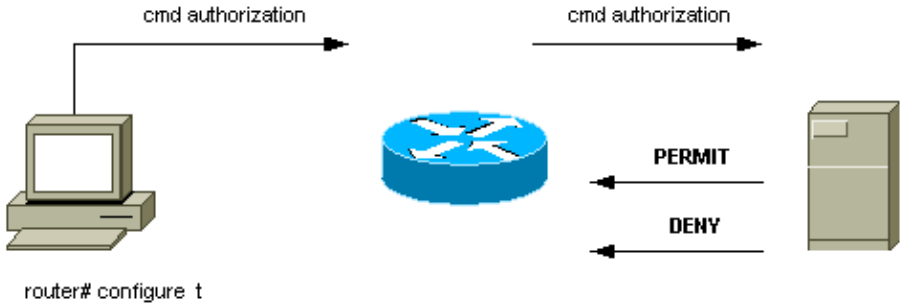
Operational Overview

The TACACS+ protocol forwards usernames and passwords to the centralized server, encrypted over the network using **MD5** one−way hashing (RFC 1321). It uses TCP port 49 as its transport protocol; this offers the following advantages over UDP (used by RADIUS):

- Connection oriented transport
- Separate acknowledgement that a request has been received (TCP ACK), regardless of how loaded the backend authentication mechanism might be
- Immediate indication of a server crash (RST packets)

During a session, if additional authorization checking is needed, the switch checks with TACACS+ to

determine if the user is granted permission to use a particular command. This provides greater control over the commands that can be executed on the switch while de−coupling from the authentication mechanism. Using command accounting, it possible to audit the commands a particular user has issued while attached to a particular network device.



When a user attempts a simple ASCII login by authenticating to a network device using TACACS+, the following process typically occurs:

- When the connection is established, the switch will contact the TACACS+ daemon to obtain a username prompt, which is then displayed to the user. The user enters a username, and the switch contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, who then enters a password that is also sent to the TACACS+ daemon.
- The network device will eventually receive one of the following responses from the TACACS+ daemon:
    - ♦ ACCEPT—The user is authenticated and service may begin. If the network device is configured to require authorization, authorization will begin at this time.
    - ♦ REJECT—The user has failed to authenticate. The user may be denied further access or will be prompted to retry the login sequence depending on the TACACS+ daemon.
    - ♦ ERROR—An error occurred at some time during authentication. This can be either at the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the network device will typically try to use an alternative method for authenticating the user.
    - ♦ CONTINUE—The user is prompted for additional authentication information.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- If TACACS+ authorization is required, the TACACS+ daemon is again contacted and returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response will contain data in the form of attributes that are used to direct the EXEC or NETWORK session for that user, determining commands that the user can access.

Recommendation

Cisco recommends the use of TACACS+, as it can be easily implemented using CiscoSecure ACS for NT, Unix, or other third−party software. TACACS+ features include detailed accounting to provide statistics on command usage and system usage, MD5 encryption algorithm, and administrative control of authentication and authorization processes.

In this example, login and enable modes use the TACACS+ server for Authentication and can fall back to local authentication if the server is unavailable. This is an important back door to leave in most networks. Use the following commands to set up TACACS+:

```
set tacacs server <server IP> primary
set tacacs server <server IP>
```

```
!--- Redundant servers are possible
set tacacs attempts 3
!--- This is the default
set tacacs key <key>
!--- MD5 encryption key
set tacacs timeout 15
!--- Longer server timeout (5 is default)
set authentication login tacacs enable
set authentication enable tacacs enable
set authentication login local enable
set authentication enable local enable
!--- The last two commands are the default; it allows fallback
to local if no TACACS+ server available
```

For more info on TACACS+ integration with CiscoSecure, see Cisco Security Technical Tips.

Other Options

It is possible to use TACACS+ authorization to control the commands each user or user−group can execute on the switch, but it is difficult to make a recommendation, as all customers have individual requirements in this area. More guidance can be found at Switch Access: Using Authentication, Authorization and Accounting.

Finally, accounting commands provide an audit trail of what each user typed and configured. Here is an example using the common practice of receiving the audit information at the end of the command:

```
set accounting connect enable start−stop tacacs+
set accounting exec enable start−stop tacacs+
set accounting system enable start−stop tacacs+
set accounting commands enable all start−stop tacacs+
set accounting update periodic 1
```

This configuration has the following features:

- The *connect* command enables accounting of outbound connection events on the switch such as telnet.
- The *exec* command enables accounting of login sessions on the switch such as operations staff.
- The *system* command enables accounting of system events on the switch such as reload or reset.
- The *commands* command enables accounting of what was entered on the switch, for both **show** and configuration commands.
- Periodic *updates* every minute to the server are helpful to record whether users are still logged in.

Configuration Checklist

Here is a summary of the recommended configurations, excluding security details.

First, remember that labeling all ports can be extremely helpful:

```
set port description <descriptive name>
```

Key:
Bold text − recommended change
Normal text − default, recommended setting

Global

Command	Comment
set vtp domain <*name*> password <*x*>	
set vtp mode transparent	
set spantree enable all	
set spantree root <*vlan*>	Position root (and secondary root)
set spantree backbonefast enable	If all switches in domain support it
set spantree uplinkfast enable	For access−layer switches only
set spantree portfast bpdu−guard enable	
set udld enable	Need port level configuration also
set test diaglevel complete	Default on Cat4000
***set test* packetbuffer sun 3:30**	Applies to Cat5000 only
set logging buffer 500	
set logging server <*IP address*>	
set logging server enable	
set logging timestamp enable	
set logging level spantree 6 default	Increase default STP syslog level
set logging level sys 6 default	Increase default System syslog level
set logging server severity 4	Allow the export of the above syslog
set logging console disable	
set snmp community read−only <*string*>	
set snmp community read−write <*string*>	
set snmp community read−write−all <*string*>	
set snmp trap enable all	
set snmp trap <*server address*> <*string*>	
set snmp rmon enable	
set ntp broadcastclient enable	
set ntp timezone <*zone name*>	
set ntp summertime <*date change details*>	
set ntp authentication enable	

set ntp key <*key*>	
set cdp enable	Also enabled on ports by default
set tacacs server <*IP address*> primary	
set tacacs server <*IP address*>	Redundant servers possible
set tacacs attempts 3	
set tacacs key <*key*>	MD5 encryption key
set tacacs timeout 15	Longer server timeout, 5 seconds is default
set authentication login tacacs enable	
set authentication enable tacacs enable	
set authentication login local enable	Default; allows fallback to local if no TACACS+ server available
set authentication enable local enable	Default; allows fallback to local if no TACACS+ server available

Host Ports

Users/mobile:

Command	Comment
set port host <*port range*>	Sets spantree portfast enable, channel off, trunk off
set udld disable <*port range*>	Disabled on copper port by default
set port speed <*port range*> auto	
set port trap <*port range*> disable	No need for SNMP traps for users

Servers:

Command	Comment
set port host <*port range*>	Sets spantree portfast enable, channel off, trunk off
set udld disable <*port range*>	Disabled on copper port by default
set port speed <port range> <10 | 100 >	
set port duplex <port range> <full | half>	Usually static/server ports, else use auto−negotiation
set port trap <port range> enable	Only key services

Unused Ports

Command	Comment
set spantree portfast *<port range>* disable	Avoid any loop problems when port next used by using full STP
set port disable *<port range>*	
set vlan *<unused dummy vlan>* *<port range>*	Direct unauthorized traffic to unused VLAN
set trunk *<port range>* off	
set port channel *<port range>* mode off	

Infrastructure Ports (switch−switch, switch−router)

Command	Comment
set udld enable *<port range>*	Not default on copper ports
set udld aggressive−mode enable *<port range>*	For devices that support it
set port negotiation *<port range>* enable	Default GE auto−negotiation
set port trap *<port range>* enable	
set trunk *<port range>* off	If not using trunks
set trunk *<mod/port>* desirable <ISL | dot1q | negotiate>	If using trunks, dot1Q is preferred.
clear trunk *<mod/port>* *<vlan range>*	Limit STP diameter by pruning
set port channel *<port range>* mode off	If not using channels
set port channel *<port range>* mode desirable	If using channels; enables PAgP
set port channel all distribution ip both	If using channels; default on Catalyst 6000
set trunk *<mod/port>* nonegotiate <ISL | dot1q>	If trunking to router, Catalyst 2900XL, 3500, or other vendor
set port negotiation *<mod/port>* disable	Necessary for some GE connections such as old software GSRs or CSS

Tools Information

For additional resources, refer to Cisco TAC Tools for LAN Technologies.

Related Information

- **RFCs, Standards & Technical Publications**
- **Internetworking Terms and Acronyms**
- **Designing Switched LAN Internetworks**
- **Multilayer LAN Switches Documentation**
- **LAN Technical Tips and Tools**
- **LAN Technology Support Pages**
- **LAN Product Support Pages**

Updated: Jul 12, 2002 Document ID: 13414