Cisco Guide to Securing Cisco NX - OS Software Devices

Guide

September 2011

Contents

Introduction **4**

Prerequisites **4**

Requirements 4
Components Used 4
Conventions 5

Principles of Secure Operations **5**

Monitor Cisco Security Advisories and Responses 5
Use Authentication, Authorization, and Accounting 5
Centralize Log Collection and Monitoring 5
Use Secure Protocols When Possible 6
Gain Traffic Visibility with NetFlow 6
Perform Configuration Management 6
Recommendations for Creating Strong Passwords 6

Securing the Management Plane **7**

General Management-Plane Hardening 8
Managing Passwords 8
Enforcing Strong Password Selection 9
Disabling Unused Services 9
Setting the EXEC Timeout Value 10
Using Management Interfaces 10
Limiting Access to the Network with Infrastructure ACLs 10
Filtering Internet Control Message Protocol Packets 11
Filtering IP Fragments 12
Securing Interactive Management Sessions 13
Encrypting Management Sessions 13
Securing the Console Port, Auxiliary Port, and Connectivity Management Processor 14
Controlling Vty Lines 14
Displaying Warning Banners 15
Using AAA 15
TACACS+ Authentication 15
Authentication Fallback 16
TACACS+ Command Authorization 16
TACACS+ Command Accounting 17
Redundant AAA Servers 17
Securing SNMP 18
SNMP Community Strings 18
SNMP Community Strings with ACLs 18
iACLs 19
SNMP Version 3 19

Logging Best Practices **20**

Send Logs to a Central Location 20
Assign Logging Level 20
Do Not Log to Console or Monitor Sessions 21
Log to the Log File 21
Configure Logging Source Interface 22
Configure Logging Time Stamps 22

Cisco NX-OS Configuration Management **22**

Configuration Checkpoint and Configuration Rollback 22
Configuration Change Notification and Logging 23

Securing the Control Plane **23**

General Control-Plane Hardening 24
IP ICMP Redirect Messages 24

ICMP Unreachable Messages 24
Proxy Address Resolution Protocol 24
NTP 24
Limiting the Effect of Control-Plane Traffic on the CPU 25
Understanding Control-Plane Traffic 25
iACLs 26
CoPP 26

Securing the Data Plane 28

General Data-Plane Hardening 28
Disabling IP Source Routing 28
Disabling ICMP Redirect Messages 28
Disabling or Limiting IP Directed Broadcasts 29
Filtering Transit Traffic with tACLs 29
Filtering ICMP Packets 29
Filtering IP Fragments 29
Implementing Antispoofing Protection 30
Configuring uRPF 30
Using IP Source Guard 31
Using Port Security 31
Using DAI 31
Configuring Antispoofing ACLs 32
Limiting the Effect of Data-Plane Traffic on the CPU 33
Features and Traffic Types That Affect the CPU 33
Traffic Identification and Traceback 34
NetFlow 34
Classification ACLs 35
Access Control with VLAN Maps and PACLs 35
Access Control with VLAN Maps 36
Access Control with PACLs 37
Access Control with MAC Address ACLs 37
Private VLANs 37
Isolated VLANs 38
Community VLANs 39
Promiscuous Ports 39

Conclusion 40

Appendix A: Cisco NX-OS Hardening Checklist 41

Appendix B: Enabling FIPS Mode 42

Introduction

This document contains information to help you secure, or harden, your Cisco® NX-OS Software system devices, which increases the overall security of your network. The document is organized according to the three planes into which functions of a network device can be categorized. It provides an overview of each security feature included in Cisco NX-OS and includes references to related documentation.

The three functional planes of a network are the management plane, control plane, and data plane. Each provides functions that need to be protected.

- Management plane: The management plane is the flow path that traffic uses when it is sent to a Cisco NX-OS device. This plane consists of applications and protocols such as Secure Shell (SSH) and Simple Network Management Protocol (SNMP).
- Control plane: The control plane of a network device processes the traffic that is important to maintaining the functions of the network infrastructure. The control plane consists of applications and protocols between network devices, including Border Gateway Protocol (BGP) and Interior Gateway Protocols (IGPs) such as Enhanced Interior Gateway Routing Protocol (EIGRP) and Open Shortest Path First (OSPF).
- Data plane: The data plane forwards data through a network device. The data plane does not include traffic that is sent to the local Cisco NX-OS device.

The discussion of security features in this document provides the essential details for engineers and administrators to configure the respective features. However, in cases where it does not, the features are explained in such a way that you can evaluate whether additional attention to a feature is required. Where possible and appropriate, this document contains recommendations that, if implemented, help secure a network.

Prerequisites

Engineers and administrators should possess a conceptual understanding of the Cisco Nexus® operating system (Cisco NX-OS) and the basic configuration options available.

Requirements

There are no specific requirements for this document.

Components Used

The guidance in this document is based on Cisco NX-OS Release 5.1. Earlier releases of Cisco NX-OS Software may not include all features or capabilities discussed here. If you are using any Cisco NX-OS release other than Release 5.1(0), please consult the release notes and documentation for that release for specific details about supported features.

This document provides guidance regarding the general capabilities of Cisco NX-OS. Specific Cisco NX-OS capabilities or feature availability may vary from platform to platform within the Cisco Nexus Family products. Not all features may be available for a specific platform. Please consult the release notes and documentation for specific hardware platforms for details regarding supported features and capabilities.

Note: Discussions of some features described in this document may refer to or use examples of options that use strong encryption algorithms. Because of U.S. government export regulations, not all encryption algorithms may be available in all releases of Cisco NX-OS in all countries.

Conventions

Refer to Cisco Technical Tips Conventions for more information about document conventions. The code lines in some command-line examples in this document are wrapped to enhance readability.

Principles of Secure Operations

Although most of this document is devoted to the secure configuration of a Cisco NX-OS device, configurations alone do not completely secure a network. The operating procedures in use on the network contribute as much to security as the configuration of the underlying devices.

This document contains operation recommendations that you are advised to implement. However, note that this document focuses on critical areas of network operations and is not comprehensive.

Monitor Cisco Security Advisories and Responses

The Cisco Product Security Incident Response Team (PSIRT) creates and maintains publications, commonly referred to as Cisco PSIRT Security Advisories, for security-related concerns in Cisco products. The method of communication used for less severe concerns is the Cisco Security Response. Security advisories and responses are available at http://www.cisco.com/go/psirt. In addition, these security advisories and responses are often complemented with Cisco Applied Mitigation Bulletins (AMBs), which provide specific details and configurations for network-based mitigation solutions for the vulnerabilities disclosed in the security advisories and responses. An extensive list of Cisco AMBs and Cisco PSIRT security advisories and responses is available on the Cisco Security Portal.

Additional information about these communication vehicles is available in the Cisco Security Vulnerability Policy.

To maintain a secure network, you must be aware of the Cisco security advisories and responses that have been released. In addition, you must obtain knowledge of a vulnerability prior to evaluating its threat to a network. Refer to Risk Triage for Security Vulnerability Announcements for assistance with this evaluation process.

Use Authentication, Authorization, and Accounting

The authentication, authorization, and accounting (AAA) framework is vital to securing network devices. The AAA framework provides authentication of management sessions, the capability to limit users to specific administrator-defined commands, and the option of logging all commands entered by all users. See the "Using AAA" section of this document for more information about AAA.

Centralize Log Collection and Monitoring

To understand existing, emerging, and historic events related to security incidents, an organization must have a unified strategy for event logging and correlation. This strategy must use logging information from all network devices and use prepackaged and customizable correlation capabilities.

After implementing centralized logging, an organization must develop a structured approach to log analysis and incident tracking. Depending on the needs of the organization, this approach can range from a simple, diligent review of log data to an advanced rule-and role-based analysis of multiple factors using correlated data.

See the "Logging Best Practices" section of this document for more information about how to implement logging on Cisco NX-OS network devices.

Use Secure Protocols When Possible

Many protocols are used to carry sensitive network management data. You must use secure protocols whenever possible. For example, use SSH instead of Telnet, so that both authentication data and management information are encrypted. In addition, you must use secure file transfer protocols when configuration data is moved or copied among the devices in a network environment. For example, use Secure Copy Protocol (SCP) instead of Trivial File Transfer Protocol (TFTP) or FTP.

See the "Securing Interactive Management Sessions" section of this document for more information about the secure management of Cisco NX-OS devices.

Gain Traffic Visibility with NetFlow

NetFlow enables engineers and administrators to monitor traffic flows throughout the network. Originally intended to export traffic information to network management applications, NetFlow can also be used to show flow information (that is, source and destination interfaces, IP addresses, and ports) on a router. This capability allows you to see traffic traversing the network in real time or to capture the information for reference. Regardless of whether flow information is exported to a remote collector or viewed live, you should configure network devices for NetFlow so that it can be used in various capacities (including proactive and reactive scenarios) if needed.

More information about this feature is available in the "Traffic Identification and Traceback" section of this document and at http://www.cisco.com/go/netflow (registered Cisco customers only).

Perform Configuration Management

Configuration management is a process by which configuration changes are proposed, reviewed, approved, and deployed. Within the context of a Cisco NX-OS device configuration, two additional aspects of configuration management are critical: configuration archival and security.

Engineers and administrators can use configuration archives to roll back changes that are made to network devices. In the context of security, configuration archives can also be used to determine what security changes were made, and when these changes occurred. In conjunction with AAA log data, this information can assist in the security auditing of network devices.

The configuration of a Cisco NX-OS device contains many sensitive details, including usernames, passwords, and the contents of ACLs (ACLs). The repository used to archive Cisco NX-OS device configurations must be secured. Insecure access to this information can undermine the security of the entire network.

Recommendations for Creating Strong Passwords

Never write passwords down, on paper or online. Instead, create passwords that you can remember easily but no one can guess easily. One way to do this is create a password that is based on a song title, affirmation, or other phrase. For example, the phrase could be "this may be one way to remember" and the password could be "TmB1w2R!" or "Tmb1W>r~" or some other variation.

Note: Do not use either of those examples as passwords.

Characteristics of a Strong Password

Strong passwords have the following characteristics:

- Contain both upper and lower case characters (e.g., a-z, A-Z).
- Contain numerals and punctuation as well as letters (e.g., 0-9, !@#$%^&*()_+|~ =\`{}[]: ;'<>?,./).
- Are at least five alphanumeric characters long.
- Are not a word in any language, and are not slang, dialect, or jargon.
- Are not based on personal information, such as the names of family members.

Characteristics of a Weak Password

A poor, weak password has the following characteristics:

- Contains fewer than eight characters.
- Is a word found in a dictionary (English or foreign).
- Is any other term that is easily guessed or found in common usage, such as:
  - The name of family, pet, friend, coworker, or fantasy character.
  - A computing term or name, such as a command, site, company, model, or application.
  - Is a birthday or another kind of personal information, such as an address or telephone number.
  - Is a predictable letter pattern or number pattern, such as aaabbb, qwerty, zyxwvuts, or 123321.
  - Any of the above, spelled backwards.
  - Any of the above, preceded or followed by a digit, such as secret1 or 1secret.

Password Security Basics

Never reveal a password.

In addition, you must:

- Never talk about a password in front of others.
- Never hint at the format of a password (such as my family name).
- Never share a password with family members.
- Never use characters from outside the standard ASCII character set. Some symbols, such the pound sterling symbol (£), are known to cause login problems on some systems.

Securing the Management Plane

The management plane consists of functions that achieve the management goals of the network. These goals include interactive management sessions using SSH, in addition to statistics gathering with tools and protocols such as SNMP or NetFlow. When considering the security of a network device, you must make sure that the management plane is protected. If a security incident undermines the functions of the management plane, recovering or stabilizing the network will be a challenge.

The following sections of this document detail the security features and configurations available in Cisco NX-OS that help fortify the management plane.

General Management-Plane Hardening

The management plane is used to access, configure, and manage a device, in addition to monitoring the device's operations and the network on which it is deployed. The management plane receives and sends traffic to support the operations of the functions listed here. Both the management and control planes of a device must be secured because the operation of these planes directly affects the overall operation of the device. The following protocols are used by the management plane:

- SNMP
- Telnet (optional)
- SSH
- FTP
- TFTP
- SCP
- TACACS+
- RADIUS
- NetFlow
- Network Time Protocol (NTP)
- Syslog

Steps must be taken to help ensure the survival of the management and control planes during security incidents. If one of these planes is successfully exploited, all planes can be compromised.

Managing Passwords

Passwords are a primary mechanism for controlling access to resources and devices. Password protection is accomplished by defining a password or secret that is used to authenticate requests. When a request is received for access to a resource or device, the request is challenged for verification (usually in the form of a request for a password and username). Access then can be granted, denied, or limited based on the authentication result. As a security best practice, passwords should be managed with a TACACS+ or RADIUS authentication server. However, note that a locally configured username and password for privileged access is still needed in the event of a TACACS+ or RADIUS service failure. Moreover, a device may also have other password information such as an NTP key, SNMP community string, or routing protocol key present within its configuration.

The **enable secret** command is used in Cisco IOS® Software to set a password that grants privileged administrative access to a Cisco IOS Software system. In Cisco NX-OS, there is no concept of an enable-secret or enable-password setting. Privileges are managed using role-based access control (RBAC). The functional equivalent to enabling mode access on a Cisco IOS Software device is assignment of an account to the network-admin (global privileged access) or vdc-admin (virtual device context [VDC]-specific privileged access) role in Cisco NX-OS.

In addition, unlike Cisco IOS Software, Cisco NX-OS does not locally store a single enable-secret cross-user shared credential as an individual password item in the configuration. Each user account maintains its own password (stored locally or through AAA), and authorization levels are dictated by the role assigned to a given account. Therefore, you must give consideration to protecting all the passwords used for all accounts assigned privileged access with the network-admin or vdc-admin roles. This task is greatly simplified if password management is centralized using AAA services.

By default, Cisco NX-OS protects all passwords used in the system configuration using irreversible MD5 hashing. There is no option to modify this behavior.

Enforcing Strong Password Selection

Cisco NX-OS has the built-in capability to optionally enforce strong password checking when a password is set or entered. This feature is enabled by default and will prevent the selection of a trivial or weak password by requiring the password to match the following criteria:

- Is at least eight characters long
- Does **not** contain many consecutive characters (abcde, lmnopq, etc.)
- Does **not** contain dictionary words (English dictionary)
- Does **not** contain many repeating characters (aaabbb, tttttyyyy, etc.)
- Does **not** contain common proper names (John, Mary, Joe, Cisco, etc.)
- Contains both uppercase and lowercase letters
- Contains numbers

If strong password checking is enabled after passwords have already been set, the system will not retroactively validate any existing passwords.

Although not recommended, password checking can be disabled using the **no password strength-checking** command or the system setup script.

For more information, refer to the "Configuring User Accounts and RBAC" section of the Cisco NX-OS Security Configuration Guide.

Disabling Unused Services

As a general security best practice, disable any unnecessary services. Cisco NX-OS does not run any of the typical Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) small servers often found in Cisco IOS Software or other network operating systems by default. As a result, these services do not need to be explicitly disabled. Cisco NX-OS is designed to **not** run remotely accessible services or protocols, by default, without explicit configuration. SSH,SNMP and NTP are essential services for running and managing a network. These services are enabled by default. If needed, they can be individually disabled. During initial setup, Cisco NX-OS will offer the option to enable Telnet. Note that this service will not load or run at boot time if it is not enabled during this initial setup. If this service is not enabled when the setup script is run, they can be added manually later if needed. Cisco recommendation is to use SSH instead of telnet for security reasons.

Cisco Discovery Protocol is a network protocol that is used to discover other devices enabled for Cisco Discovery Protocol for neighbor adjacency and to map a network topology. Cisco Discovery Protocol can be used by network management systems or during troubleshooting. Cisco Discovery Protocol is enabled by default in Cisco NX-OS. Cisco Discovery Protocol must be disabled on all interfaces that are connected to untrusted networks. This disabling is accomplished with the **no cdp enable** interface command. Alternatively, Cisco Discovery Protocol can be disabled globally with the **no cdp enable** global configuration command. Note that Cisco Discovery Protocol can be exploited by malicious users, or reconnaissance and network mapping.

Link Layer Discovery Protocol (LLDP) is an IEEE protocol defined in the IEEE 802.1AB standard. LLDP is similar to Cisco Discovery Protocol; however, this protocol allows interoperability between devices not supported by the Cisco Discovery Protocol. By default, LLDP is not enabled in Cisco NX-OS. To enable it, the feature set must be enabled using the **feature lldp** global configuration command. When enabled, LLDP must be treated in the same manner as Cisco Discovery Protocol and disabled on all interfaces that connect to untrusted networks.

To accomplish this, run the **no lldp transmit** and **no lldp receive** interface configuration commands. Run the **no feature lldp** configuration command to disable LLDP globally. Like Cisco Discovery Protocol, LLDP has the potential to be exploited by malicious users for reconnaissance and network mapping.

Setting the EXEC Timeout Value

To set the interval that the EXEC command interpreter waits for user input before it terminates a session, run the **exec-timeout** line configuration command. The **exec-timeout** command must be used to log out sessions on a vty or physical terminal line (tty) that is left idle (inactive). By default in Cisco NX-OS, sessions are set to disconnect after 30 minutes of inactivity.

```
!
line console
 exec-timeout <minutes>
line vty
 exec-timeout <minutes>
!
```

Using Management Interfaces

The management plane of a device can be accessed in-band or out-of-band on a physical or logical management interface. Ideally, both in-band and out-of-band management access exist to provide redundancy, so that the management plane can be accessed in the event of a network outage.

One of the most common interfaces used for in-band access to a device is the loopback interface. Loopback interfaces are logical; therefore, they are always up, whereas physical interfaces can change state, making the interface potentially inaccessible. You should add a loopback interface as a management interface to each device. This interface should be used exclusively for the management plane. This approach allows the administrator to apply policies throughout the network for the management plane. After the loopback interface is configured on a device, it can be used by management plane protocols such as SSH, SNMP, and syslog to send and receive traffic.

Depending on the Cisco NX-OS platform, a dedicated management interface may be available, as is the case on the Cisco Nexus 7000 Series Switches. In these cases, the physical management interface can be used to access the logical management interfaces of the device. Typically, this physical management interface is isolated for access through the use of Virtual Route Forwarding (VRF) tables, with the default management VRF associated with the physical management interface. By restricting management traffic to the management VRF using ACLs, a very effective side-band or out-of-band management topology can be established.

Limiting Access to the Network with Infrastructure ACLs

Devised to prevent unauthorized direct communication to network devices, infrastructure ACLs (iACLs) are one of the most critical security controls that can be implemented in networks. iACLs use the idea that nearly all network traffic simply traverses the network and is not destined for the network itself.

The key to an iACL is its construction. iACLs are built on the premise of permitting connections among trusted hosts or networks that require communication with network infrastructure devices according to established security policies and configurations. This required communication typically consists of management- and control-plane traffic. Common examples of these types of connections are external BGP (eBGP), SSH, and SNMP. After the required connections have been permitted, all other traffic to the infrastructure is explicitly denied.

All transit traffic that crosses the network and is not destined for infrastructure devices is then explicitly permitted (this permission typically occurs through a transit ACL [tACL], discussed later in this document).

The protections provided by iACLs are relevant to both the management and control planes. The implementation of iACLs can be made easier through the use of distinct addressing for network infrastructure devices. Refer to the Cisco white paper A Security-Oriented Approach to IP Addressing for more information about the security implications of IP addressing.

This example iACL configuration illustrates a structure that can be used as a starting point when beginning the iACL implementation process:

```
!
ip access-list extended ACL-INFRASTRUCTURE-IN
! !--- Permit required connections for routing protocols and
! !--- network management
   permit tcp host <trusted-ebgp-peer> host <local-ebgp-address> eq 179
   permit tcp host <trusted-ebgp-peer> eq 179 host <local-ebgp-address>
   permit tcp host <trusted-management-stations> any eq 22
   permit udp host <trusted-netmgmt-servers> any eq 161
! !--- Deny all other IP traffic to any network device !
   deny ip any <infrastructure-address-space> <mask>
! !--- Permit transit traffic !
   permit ip any any
!
```

For the strongest protection of infrastructure devices, deployed iACLS should be applied in the ingress direction on all interfaces for which an IP address has been configured, including interfaces that connect to other organizations, remote-access segments, user segments, and segments in data centers. Note an iACL cannot provide complete protection against vulnerabilities when the attack originates from a trusted source address.

The use of Cisco NX-OS port profiles can greatly simplify the deployment and maintenance of ACLs, including iACLs.

Refer to the Cisco white paper Protecting Your Core: Infrastructure Protection Access Control Lists for more information about iACLs.

Filtering Internet Control Message Protocol Packets

The Internet Control Message Protocol (ICMP) was designed as an IP control protocol. As such, the messages it conveys can have far-reaching ramifications for TCP and IP in general. Although the network troubleshooting tools **ping** and **traceroute** use ICMP, external ICMP connectivity is rarely needed for the proper operation of a network.

Cisco NX-OS provides functions to specifically filter ICMP messages by name or type and code. This example ACL, used with access control entries from the previous examples, allows pings from trusted management stations and network management system servers while blocking all other ICMP packets:

```
!
ip access-list extended ACL-INFRASTRUCTURE-IN
! !--- Permit ICMP Echo (ping) from trusted management stations and servers !
 permit icmp host <trusted-management-stations> any echo
 permit icmp host <trusted-netmgmt-servers> any echo
! !--- Deny all other IP traffic to any network device !
 deny ip any <infrastructure-address-space> <mask>
! !--- Permit transit traffic !
 permit ip any any
!
```

Filtering IP Fragments

The filtering of fragmented IP packets can pose a challenge to infrastructure and security devices alike. This challenge exists because the Layer 4 information that is used to filter TCP and UDP packets is present only in the initial fragment. Cisco NX-OS uses a specific method to check non-initial fragments against configured ACLs. Cisco NX-OS evaluates these non-initial fragments against the ACL and ignores any Layer 4 filtering information. This approach causes non-initial fragments to be evaluated solely on the Layer 3 portion of any configured access control entry.

In this example configuration, if a TCP packet destined for 192.168.1.1 on port 22 is fragmented in transit, the initial fragment is dropped as expected by the second access control entry based on the Layer 4 information within the packet. However, all remaining (non-initial) fragments are allowed by the first access control entry, based completely on the Layer 3 information in the packet and the access control entry rules. This scenario is shown in the following configuration:

```
!
ip access-list extended ACL-FRAGMENT-EXAMPLE
   permit tcp any host 192.168.1.1 eq 80
   deny tcp any host 192.168.1.1 eq 22
!
```

Due to this non-intuitive nature of fragment handling, IP fragments are often inadvertently permitted by ACLs. Moreover, fragmentation is often used in attempts to evade detection by intrusion-detection systems. For these reasons, IP fragments are often used in attacks, and so they must be explicitly filtered at the top of any configured iACLs. This example ACL includes comprehensive filtering of IP fragments. The functions in this example should be used in conjunction with the functions in the previous examples.

```
!
ip access-list extended ACL-INFRASTRUCTURE-IN
! !--- Deny IP fragments using protocol-specific ACEs to aid in
! !--- the classification of attack traffic !
   deny tcp any any fragments
   deny udp any any fragments
   deny icmp any any fragments
   deny ip any any fragments
! !--- Deny all other IP traffic to any network device !
    deny ip any <infrastructure-address-space> <mask>
! !--- Permit transit traffic !
```

```
        permit ip any any
    !
```

Refer to the Cisco white paper Access Control Lists and IP Fragments for more information about ACL handling of fragmented IP packets.

Securing Interactive Management Sessions

Management sessions for devices allow you to view and collect information about a device and its operations. If this information is disclosed to a malicious user, the device can become the target of an attack, compromised, and commandeered to perform additional attacks. Anyone with privileged access to a device has the capability for full administrative control of that device. Securing management sessions is imperative to prevent information disclosure and unauthorized access.

Encrypting Management Sessions

Because information can be disclosed during an interactive management session, this traffic must be encrypted so that a malicious user cannot gain access to the data being transmitted. Encrypting the traffic allows a secure remote-access connection to the device. If the traffic for a management session is sent over the network in clear text, an attacker can obtain sensitive information about the device and the network.

An administrator can establish an encrypted and secure remote access management connection to a device by using SSH. Cisco NX-OS supports SSH Version 2.0 (SSHv2) only. Note that SSHv1 and v2 are not compatible. As of NX-OS Release 5.1, SSH also runs in FIPS mode. For more information, consult the Cisco NX-OS SSH configuration guide and documentation.

Cisco NX-OS also supports SCP and Secure FTP (SFTP), which allow an encrypted and secure connection for copying device configurations or software images. SCP relies on SSH. This example configuration enables SSH on a Cisco NX-OS device:

```
!
ip domain-name example.com
!
feature ssh
ssh key rsa 2048
!
ssh login-attempts <1-10> (default is 3)
!
```

This configuration example enables SCP and SFTP services:

```
!
feature scp-server
feature sftp-server
!
```

Refer to the “Configuring SSH and Telnet” section of the Cisco Nexus 7000 Series NX-OS Security Configuration Guide for more information about the Cisco NX-OS SSH, SCP, and SFTP features.

Securing the Console Port, Auxiliary Port, and Connectivity Management Processor

In Cisco NX-OS devices, console and auxiliary (AUX) ports are asynchronous lines that can be used for local and remote access to a device. You must be aware that console ports on Cisco NX-OS devices have special privileges. In particular, these privileges allow an administrator to perform the password recovery procedure. To perform password recovery, an unauthenticated attacker would need to have access to the console port and the capability to interrupt power to the device or to cause the device to fail.

Any method used to access the console port of a device must be secured with a security level that is equal to the security that is enforced for privileged access to a device. The configuration of AAA authentication methods and policies applied to the login mechanism will automatically apply to the console, AUX port, and vty access methods.

The AUX port (also called com1), when available, cannot be explicitly disabled. Therefore, AAA must be properly configured globally on the platform to secure the AUX port as well. In addition, it is highly recommended that physical security measures be applied to restrict physical access to the AUX port.

Some Cisco NX-OS platforms provide an optional connectivity management processor (CMP) for side-band or out-of-band access to the console. The CMP functions internally as an independent device, much like an integrated lights-out (iLO) port on a server or a built-in terminal server. The CMP runs an independent OS (a reduced version of Cisco NX-OS) on an independent system processor. Authentication for the CMP is tied to the AAA methods for authentication configured on the main system supervisor. If the configured AAA servers can be reached through the main supervisor, the CMP will authenticate using the configured AAA policies and methods. If the AAA server is not available, the CMP will use local authentication, checking against a user database stored locally on the CMP.

To adequately secure the CMP (if it is used), AAA should be configured on the main system supervisor. and the CMP local authentication database should be set up with an individual administrative password.

The CMP is accessed over an IP network using the SSH protocol. If the CMP is not going to be used, it can be disabled simply by not assigning an IP address to it or by removing the IP address from the CMP interface if one is already assigned.

Controlling Vty Lines

Interactive management sessions in Cisco NX-OS use a virtual tty (vty). A vty line is used for all remote network connections supported by the device, regardless of protocol (SSH, SCP, or Telnet are examples). To help ensure that a device can be accessed through a local or remote management session, proper controls must be enforced on vty lines. Cisco NX-OS devices have a limited number of vty lines; the number of configured lines available can be determined by using the **show run vshd** command. By default, up to 16 concurrent vty sessions are allowed. When all vty lines are in use, new management sessions cannot be established, creating a denial-of-service (DoS) condition for access to the device.

The simplest form of access control for the vty of a device is the use of authentication on all lines regardless of the device location within the network. Vty access controls can be enforced by using the **access-class** configuration commands, using the control-plane policing (CoPP) feature, or applying access lists to interfaces on the device.

Authentication can be enforced using the local user database or through the use of AAA, which is the recommended method for authenticated access to a device.

The **exec-timeout** command must be used to log out sessions on any vty that is left idle.

Vty lines in Cisco NX-OS automatically accept connections using any configured transport protocols. To disable a specific (for instance, unsecured) protocol from accessing vty sessions, you must globally disable the specific protocol. For example, to prevent a Telnet session to the vty line, you must disable Telnet globally using the **no feature telnet** command.

Displaying Warning Banners

In some legal jurisdictions, you cannot prosecute or legally monitor malicious users unless they have been notified that they are not permitted to use the system. One way to provide this notification is to place this information in a banner message that is configured with the Cisco NX-OS banner login command.

Legal notification requirements are complex and vary by jurisdiction and situation and should be discussed with legal counsel. Even within jurisdictions, legal opinions can differ. In cooperation with counsel, a banner can provide some or all of the following information:

- Notice that the system is to be logged into or used only by specifically authorized personnel, and perhaps information about who can authorize use
- Notice that any unauthorized use of the system is unlawful and can be subject to civil and criminal penalties
- Notice that any use of the system can be logged or monitored without further notice, and that the resulting logs can be used as evidence in court
- Specific notices required by local laws

From a security (rather than legal) point of view, a login banner should not contain any specific information about the router name, model, software, or ownership. This information can be abused by malicious users.

Using AAA

The AAA framework is critical to securing interactive access to network devices. The AAA framework provides a highly configurable environment that can be tailored depending on the needs of the network.

TACACS+ Authentication

TACACS+ is an authentication protocol that Cisco NX-OS devices can use for authentication of management users against a remote AAA server. These management users can access the Cisco NX-OS device through SSH or Telnet.

TACACS+ authentication, or more generally AAA authentication, provides the capability to centralize authentication information and authorization policies. It also enables effective centralized accounting of AAA-related transactions for improved auditability.

RADIUS is a protocol similar in purpose to TACACS+; however, RADIUS encrypts only the password sent across the network. In contrast, TACACS+ encrypts the entire TCP payload, including both the username and password. For this reason, TACACS+ is preferred over RADIUS when TACACS+ is supported by the AAA server and network device. Refer to "TACACS+ and RADIUS Comparison" design technote for a more detailed comparison of these two protocols.

TACACS+ authentication can be enabled on a Cisco NX-OS device using a configuration similar to this example:

```
!
! TACACS+ must be enabled in NX-OS
feature tacacs+
aaa authentication login default group tacacs+
!
tacacs-server host <ip-address-of-tacacs-server>
tacacs-server key <key>
!
```

The previous configuration can be used as a starting point for an organization-specific AAA authentication template. Refer to the "Use Authentication, Authorization, and Accounting" section of this document for more information about the configuration of AAA.

Authentication Fallback

If all configured AAA servers become unavailable, then a Cisco NX-OS device can rely on secondary authentication methods. Configuration options include the use of local or no authentication if all configured TACACS+ servers are unavailable. You should not use the None option, which in effect would fall back to no authentication if the AAA servers are unreachable. This fallback would potentially allow a DoS attack on the AAA servers to eliminate authentication on the network devices. Instead, authentication fallback should be set to use the local database when AAA servers are unreachable. This approach allows a locally defined user to be created for one or more network administrators. If TACACS+ were to become completely unavailable, each administrator can use his or her local username and password.

Although this action does enhance the accountability of network administrators during TACACS+ outages, it can increase the administrative overhead since local user accounts on all network devices must be maintained. Some of this overhead can be reduced by using the Cisco NX-OS TACACS+ configuration distribution mechanism, which uses the Cisco Fabric Services protocol.

This configuration example builds on the previous TACACS+ authentication example, including fallback authentication to the password that is configured locally with the **enable secret** command:

```
!
username admin password <password> role network-admin
!
aaa authentication login default group tacacs+
aaa authentication login default fallback error local
!
```

Refer to "Configuring Authentication" for more information about the use of fallback authentication with AAA.

TACACS+ Command Authorization

Command authorization with TACACS+ and AAA provides a mechanism that permits or denies each command that is entered by an administrative user. When the user enters EXEC or configuration commands, Cisco NX-OS sends each command to the configured AAA server. The AAA server then uses its configured policies to permit or deny the command for that particular user.

This configuration can be added to the previous AAA authentication example to implement command authorization:

```
!
aaa authorization commands default group <server group> [local]
aaa authorization config-commands default group <server group> [local]
!
```

Refer to the "Configuring AAA" section in the Cisco NX-OS Security Configuration Guide for more information about command authorization.

TACACS+ Command Accounting

When configured, AAA command accounting sends information about each EXEC or configuration command that is entered back to the configured TACACS+ servers. The information sent to the TACACS+ servers includes the command executed, the date it was executed, and the username of the user entering the command. Command accounting is not supported using RADIUS.

This example configuration enables AAA command accounting for all commands entered. This configuration builds on previous examples that include configuration of the TACACS servers.

```
!
aaa accounting default group <server group>
!
```

Refer to the "Configuring AAA" section in the Cisco NX-OS Security Configuration Guide for more information regarding the configuration of AAA accounting.

Redundant AAA Servers

The AAA servers that are used in an environment should be redundant and deployed in a fault-tolerant manner. This approach helps ensure that interactive management access, such as SSH access, is possible if an AAA server is unavailable.

When you design or implement a redundant AAA server solution, keep these considerations in mind:

- Availability of AAA servers during potential network failures
- Geographically dispersed placement of AAA servers
- Load on individual AAA servers during steady-state and failure conditions
- Network latency between network access servers and AAA servers
- AAA server databases synchronization

Cisco NX-OS supports the configuration of server groups for redundant AAA services for both RADIUS and TACACS+. It is highly recommended that this facility be used to provide AAA service robustness. The following is an example of TACACS+ server group configuration for redundant AAA servers:

```
!
tacacs-server host <tacacs+ server1 IP>
tacacs-server host <tacacs+ server2 IP>
tacacs-server host <tacacs+ server3 IP>
! Global key for all TACACS+ servers (you can optionally define individual keys
per server)
tacacs-server key <key>
!
```

```
aaa authentication group <group name>
   server <tacacs+ server1 IP>
   server <tacacs+ server2 IP>
   server <tacacs+ server3 IP>
!
```

Refer to the "Configuring TACACS+" or "Configuring RADIUS" section in the "Cisco NX-OS Security Configuration Guide" for more information about AAA server groups.

Securing SNMP

This section discusses several methods that can be used to secure the deployment of SNMP in Cisco NX-OS devices. SNMP must be properly secured to protect the confidentiality, integrity, and availability of both the network data and the network devices through which this data transits. SNMP provides a wealth of information about the health of network devices. This information should be protected from malicious users who want to use this data for attacks against the network.

SNMP Community Strings

Community strings are passwords that are applied to a Cisco NX-OS device to restrict access, both read-only and read-write access, to the SNMP data on the device. These community strings, as with all passwords, should be carefully chosen to help ensure that they are strong. Community strings should be changed at regular intervals and in accordance with network security policies. For example, the strings should be changed when a network administrator changes roles or leaves the company.

These configuration lines configure a read-only community string of READONLY and a read-write community string of READWRITE:

```
!
snmp-server community READONLY ro
snmp-server community READWRITE rw
!
```

Note that the preceding community string examples have been chosen to clearly explain the use of these strings. As with any other passwords used for production environments, community strings should be chosen with caution and should consist of a series of alphabetical, numerical, and nonalphanumeric symbols that are not easily guessed or compromised using dictionary attacks.

Refer to the Recommendations for Creating Strong Passwords section for more information about the selection and generation of strong passwords.

Refer to the Cisco NX-OS SNMP Command Reference for more information about this feature.

SNMP Community Strings with ACLs

In addition to the community string, an ACL should be applied that further restricts SNMP access to a selected group of source IP addresses. This configuration restricts SNMP read-only access to end host devices that reside in the 192.168.100.0/24 address space, and it restricts SNMP read-write access to only the end host device at 192.168.100.1.

Note that the devices permitted by these ACLs require the proper community string to access the requested SNMP information.

```
!
ip access-list allow_snmp_ro
   permit ip 192.168.100.0/24 any
!
ip access-list allow_snmp_rw
   permit ip host 192.168.100.1 any
!
snmp-server community READONLY use-acl allow_snmp_ro
snmp-server community readwrite use-acl allow_snmp_rw
!
```

Refer to the "Configuring SNMP" section of the Cisco NX-OS System Management Configuration Guide for more details.

iACLs

iACLs can be deployed to help ensure that only end hosts with trusted IP addresses can send SNMP traffic to a Cisco NX-OS device. An iACL should contain a policy that denies unauthorized SNMP packets on UDP port 161.

See the "Limiting Access to the Network with Infrastructure ACLs" section of this document for more information about the use of iACLs.

SNMP Version 3

SNMP Version 3 (SNMPv3) is defined by RFC3410, RFC3411, RFC3412, RFC3413, RFC3414, and RFC3415 and is an interoperable standards-based protocol for network management. SNMPv3 provides secure access to devices by authenticating and optionally encrypting packets over the network. Where supported, SNMPv3 can be used to add another layer of security when deploying SNMP. SNMPv3 consists of three primary configuration options:

- no auth: This mode does not require any authentication nor any encryption of SNMP packets
- auth: This mode requires authentication of the SNMP packet without encryption
- priv: This mode requires both authentication and encryption (privacy) of each SNMP packet

An authoritative engine ID must exist to use the SNMPv3 security mechanism's authentication or authentication and encryption to handle SNMP packets; by default, the engine ID is generated locally. The engine ID can be displayed with the **show snmp engineID** command as shown in this example:

```
Nexus7000-Lab# show snmp engineID
Local SNMP engineID: [Hex] 8000000903001B54C24100
                     [Dec] 128:000:000:009:003:000:027:084:194:065:000
```

Note that if the engine ID is changed, all SNMP user accounts must be reconfigured.

SNMPv3 is enabled by default in Cisco NX-OS and cannot be explicitly disabled.

To access a Cisco NX-OS device using SNMPv3, a user or administrator must have a valid SNMP account. SNMP user accounts can be explicitly created and are also automatically generated by the system to synchronize with valid accounts verified through local or AAA-based authentication.

For more information about Cisco NX-OS automatic user synchronization, see the "Configuring SNMP" section of the Cisco NX-OS System Management Configuration Guide.

You can explicitly configure SNMP to require message authentication and encryption for incoming requests. By default, the SNMP agent in Cisco NX-OS accepts SNMPv3 messages without authentication and encryption. It is recommended that authentication and encryption be required and enforced for SNMP v3 messages.

The following global configuration command enforces SNMP message encryption for all users:

```
!
snmp-server globalEnforcePriv
!
```

This command explicitly configures the SNMPv3 user **snmpv3user** with an MD5 authentication password of **authpassword** and a AES-128 encryption password of **privpassword**:

```
!
snmp-server user snmpv3user auth sha authpassword priv aes-128 privpassword
!
```

Refer to the "Configuring SNMP" section of the Cisco NX-OS System Management Configuration Guide for more information about configuring SNMPv3.

Logging Best Practices

Event logging gives you visibility into the operation of a Cisco NX-OS device and the network in which it is deployed. Cisco NX-OS provides several flexible logging options that can help achieve the network management and visibility goals of an organization.

The following sections provide some basic logging best practices that can help an administrator use logging successfully while reducing the impact of logging on a Cisco NX-OS device.

Send Logs to a Central Location

You should send logging information to a remote syslog server. By doing so, you can correlate and audit network and security events across network devices more effectively. Note that syslog messages are transmitted unreliably by UDP and in cleartext. For this reason, any protections that a network offers to management traffic (for example, encryption and out-of-band access) should be extended to include syslog traffic.

This example configures a Cisco NX-OS device to send logging information to a remote syslog server using the management VRF instance:

```
!
logging server <ip-address|hostname> use-vrf management
!
```

Assign Logging Level

Each internal system software component of Cisco NX-OS that is capable of logging using the syslog facility can be assigned one of eight severity levels that range from level 0, Emergencies, through level 7, Debug. The severity level chosen will determine the level, granularity, and frequency of messages generated for that component. Unless specifically required, you are advised to avoid logging at level 7. Logging at level 7 produces an elevated CPU load on the device that can lead to device and network instability.

This configuration example limits log messages that are sent to remote syslog servers to severity levels 6 (informational) through 0 (emergencies):

```
!
logging server <ip-address|hostname> 6 use-vrf management
!
```

Refer to "Configuring System Message Logging" in the Cisco NX-OS System Management Configuration Guide for more information about remote logging configuration.

Do Not Log to Console or Monitor Sessions

With Cisco NX-OS, you can send log messages to monitor sessions, or to the console. However, doing so can elevate the CPU load of a Cisco NX-OS device, and therefore is not recommended. Furthermore, you are advised to send logging information to the local log buffer or the local log file, which can be viewed using the **show logging** command.

Use the global configuration commands **no logging console** and **no logging monitor** to disable logging to the console and to monitor sessions. This configuration example shows the use of these commands:

```
!
no logging console
no logging monitor
!
```

If logging output is required for troubleshooting purposes, you should enable it only temporarily, to monitor for vty sessions, and avoid using it on the console. Be sure to disable logging to monitor sessions after troubleshooting is completed.

Refer to the Cisco NX-OS System Management Configuration Guide for more information about global configuration commands for logging.

Log to the Log File

Cisco NX-OS software supports the use of a local log buffer in the form of a log file so that an administrator can view locally generated log messages. The use of buffered logging to the log file is highly recommended instead of logging to either the console or monitor sessions.

By default, the log file is stored on the storage media in the logflash slot, which is represented by **logflash: device** in Cisco NX-OS at the command-line interface (CLI). The default filename for the log file is **messages**, which is the standard UNIX **logging file**. Using logging logfile command, you can change the name of the log file, but the location of the log file (**logflash:**) cannot be altered.

There are two configuration options that are relevant when configuring buffered logging: the logging buffer size and the message severity levels stored in the buffer. The size of the log file and the severity levels of messages sent to the log file can be configured using the **logging logfile** global command. An administrator can view the contents of the logging buffer through the **show logging** EXEC command.

This configuration example sets the size of the log file to 16384 bytes and the severity level to 6, informational, indicating that messages at levels 0 (emergencies) through 6 (informational) are stored. The default name of the log file has been kept:

```
!
logging logfile messages 6 size 16384
!
```

Refer to the Cisco NX-OS System Management Configuration Guide for more information about buffered logging to a log file.

Configure Logging Source Interface

To provide an increased level of consistency when collecting and reviewing log messages, you should statically configure a **logging source interface**. Accomplished through the logging source-interface interface command, statically configuring a logging source interface helps ensure that the same IP address appears in all logging messages that are sent from an individual Cisco NX-OS device. For added stability, you should use a loopback interface as the logging source.

This configuration example illustrates the use of the **logging source-interface interface** global configuration command to specify that the IP address of the loopback 0 interface should be used for all log messages:

```
!
logging source-interface Loopback 0
!
```

Refer to the Cisco NX-OS System Management Configuration Guide for more information.

Configure Logging Time Stamps

The configuration of logging time stamps helps you correlate events across network devices. It is important to implement a correct and consistent logging time-stamp configuration to help ensure that you can correlate logging data. Logging time stamps should be configured to include millisecond precision.

This example includes the configuration of logging time stamps with millisecond precision:

```
!
logging timestamp milliseconds
!
```

Cisco NX-OS logging will automatically time stamp log entries with the date and time in the locally configured time zone of the device.

Cisco NX-OS Configuration Management

Cisco NX-OS includes several features that can enable a form of configuration management on a Cisco NX-OS device. Such features include functions to archive configurations and to roll back a configuration to a previous version and create a detailed configuration change log.

Configuration Checkpoint and Configuration Rollback

Cisco NX-OS provides an integrated facility for generating configuration checkpoints. This feature allows the system to maintain an archive of snapshot configurations. The configuration of the device can be rolled back to any of the archived configuration checkpoints at any time by an administrator.

A manual configuration checkpoint can be initiated with the **checkpoint** command. Automated configuration checkpoints can be generated periodically by combining the checkpoint and scheduler features of Cisco NX-OS. The following configuration creates a scheduler job to automatically generate a configuration checkpoint every eight hours:

```
!
feature scheduler
!
scheduler job name auto_checkpoint
   checkpoint
   end-job
!
scheduler schedule name 8hr_checkpoint
   job name auto_checkpoint
   time start now repeat 00:08:00
!
```

Checkpoints in the internal system checkpoint database can be viewed with the command **show checkpoint summary**, and the actual contents of the checkpoint files can be viewed with **show checkpoint**.

A running configuration can be rolled back to a checkpoint using the **rollback** command.

Refer to the "Configuring Rollback" and "Configuring the Scheduler" sections of the Cisco NX-OS System Management Configuration Guide for further information about these features.

Configuration Change Notification and Logging

Cisco NX-OS can log configuration change events along with the individual changes when AAA command accounting is enabled.

With command accounting enabled, all CLI commands entered, including configuration commands, are logged to the configured AAA server. Using this information, a forensic trail for configuration change events along with the individual commands entered for those changes can be recorded and reviewed.

Because of this capability, it is strongly advised that AAA command accounting be enabled and configured.

Refer to the "TACACS+ Command Accounting" section of this document for more information.

Securing the Control Plane

Control-plane functions consist of the protocols and processes that communicate between network devices to move data from the source to the destination. These include routing protocols such as BGP, as well as protocols such as ICMP.

It is important that events in the management and data plane do not adversely affect the control plane. If a data plane event such as a DoS attack affects the control plane, the entire network can become unstable. The information in this section about Cisco NX-OS features and configurations can help ensure the resilience of the control plane.

General Control-Plane Hardening

Protecting the control plane of a network device is critical because the control plane helps ensure that the management and data planes are maintained and operational. If the control plane were to become unstable during a security incident, it can be impossible to recover the stability of the network.

In many cases, disabling the reception and transmission of certain types of messages on an interface can reduce the CPU load that is required to process unneeded packets.

IP ICMP Redirect Messages

An ICMP redirect message can be generated by a router when a packet is received and transmitted on the same interface. In this situation, the router forwards the packet and sends an ICMP redirect message back to the sender of the original packet. This behavior allows the sender to bypass the router and forward future packets directly to the destination (or to a router closer to the destination). In a properly functioning IP network, a router sends redirect messages only to hosts on its own local subnets. In other words, ICMP redirect messages should never go beyond a Layer 3 boundary.

There are two types of ICMP redirect messages: redirect messages for a host address, and redirect messages for an entire subnet. A malicious user can exploit the capability of the router to send ICMP redirect messages by continually sending packets to the router, forcing the router to respond with ICMP redirect messages, resulting in adverse impact on the CPU and on the performance of the router. To prevent the router from sending ICMP redirect messages, use the **no ip redirects** interface configuration command.

Refer to the Cisco document titled, "Configuring IP Services" for more information about ICMP redirects.

ICMP Unreachable Messages

Filtering with an interface access list elicits the transmission of ICMP unreachable messages back to the source of the filtered traffic. Generating these messages can increase CPU utilization on the device. You can disable ICMP unreachable message generation using the interface configuration command **no ip unreachables**.

Proxy Address Resolution Protocol

Proxy Address Resolution Protocol (ARP) is the technique in which one device, usually a router, answers ARP requests that are intended for another device. By "faking" its identity, the router accepts responsibility for routing packets to the real destination. Proxy ARP can help machines on a subnet reach remote subnets without the need to configure routing or a default gateway. Proxy ARP is defined in RFC 1027.

There are several disadvantages to using proxy ARP. Proxy ARP can result in an increase in the amount of ARP traffic on the network segment and resource exhaustion and man-in-the-middle attacks. Proxy ARP presents a resource exhaustion attack vector because each proxied ARP request consumes a small amount of memory. An attacker could attempt to exhaust memory unnecessarily by sending a large number of ARP requests.

Man-in-the-middle attacks enable a host on the network to spoof the MAC address of the router, causing unsuspecting hosts to send traffic to the attacker. Proxy ARP can be disabled using the interface configuration command **no ip proxy-arp**.

NTP

NTP is not an especially dangerous service, but any unneeded service can represent an attack vector. If NTP is used, you should be sure to explicitly configure a trusted time source and to use proper authentication. Accurate and reliable time can be very useful for logging purposes, such as for forensic investigations of potential attacks.

Configuring NTP authentication provides assurance that NTP messages are exchanged between trusted NTP peers. You should enable authentication for NTP if at all possible. Additionally, for precision and redundancy purposes, you should configure multiple NTP server time sources on the Cisco NX-OS device acting as an NTP client.

For more information about configuring NTP, including enabling NTP authentication, please refer to the "Configuring NTP" section of the Cisco NX-OS System Management Configuration Guide.

Limiting the Effect of Control-Plane Traffic on the CPU

Protection of the control plane is critical. Because application performance and end-user experience can suffer without the presence of data and management traffic, the survivability of the control plane helps ensure that the other two planes are maintained and operational.

Understanding Control-Plane Traffic

To properly protect the control plane of the Cisco NX-OS device, you must understand the types of traffic that are process switched by the CPU. Process-switched traffic normally consists of two types of traffic. The first type of traffic is directed to the Cisco NX-OS device and must be handled directly by the Cisco NX-OS device CPU. This traffic consists of this category:

- Receive adjacency traffic: This traffic contains an entry in the Cisco Express Forwarding table through which the next router hop is the device itself, which is indicated by the term **receive** in the **show ip cef** CLI output. This indication appears for any IP address that requires direct handling by the Cisco NX-OS device CPU, including interface IP addresses, multicast address space, and broadcast address space.

The second type of traffic that is handled by the CPU is data-plane traffic with a destination beyond the Cisco NX-OS device itself that requires special processing by the CPU. This type of behavior tends to be platform specific and dependent on the specific hardware implementation of the specific Cisco NX-OS platform. Some platforms handle more types of data-plane traffic in hardware, thereby requiring less CPU-based intervention. Regardless of the hardware handling capabilities, you should understand potential sources of control-plane traffic that could affect the system CPU. Although not an exhaustive list of data-plane traffic that can affect the CPU, these types of traffic are potentially process switched and can therefore affect the operation of the control plane:

- ACL logging: ACL logging traffic consists of any packets that are generated due to a match (permit or deny) of an access control entry on which the **log** keyword is used.
- Unicast Reverse Path Forwarding (uRPF): uRPF, used in conjunction with an ACL, can result in the process switching of certain packets.
- IP options: Any IP packets with options included must be processed by the CPU.
- Fragmentation: Any IP packet that requires fragmentation must be passed to the CPU for processing.
- Time-to-live (TTL) expiry: Packets that have a TTL value less than or equal to 1 require ICMP Time Exceeded (ICMP Type 11, Code 0) messages to be sent, which results in CPU processing.
- ICMP unreachable messages: Packets that result in ICMP unreachable messages due to routing, maximum transmission unit (MTU), or filtering are processed by the CPU.
- Traffic requiring an ARP request: Destinations for which an ARP entry does not exist require processing by the CPU.
- Non-IP traffic: All non-IP traffic is processed by the CPU.

The following list details several methods to determine which types of traffic are being processed by the Cisco NX-OS device CPU:

- The **show ip cef** command provides the next-hop information for each IP prefix that is contained in the Cisco Express Forwarding table. As indicated previously, entries that contain **receive** as the next hop are considered receive adjacencies and indicate that traffic must be sent directly to the CPU.
- The **show interface switching** command provides information about the number of packets being process switched by a device.
- The **show ip traffic** command provides information about the number of IP packets:
  - With a local destination (that is, receive adjacency traffic)
  - With options
  - That require fragmentation
  - That are sent to broadcast address space
  - That are sent to multicast address space

Receive adjacency traffic can be identified through the use of the **show ip cache flow** command. Any flows that are destined for the Cisco NX-OS device have a destination interface (DstIf) of **local**.

CoPP can be used to identify the type and rate of traffic that reaches the control plane of the Cisco NX-OS device. CoPP can be performed through the use of granular classification ACLs, logging, and the **show policy-map control-plane** command.

iACLs

iACLs limit external communication to the devices of the network. iACLs are extensively covered in the "Limiting Access to the Network with Infrastructure ACLs" section of this document.

You should implement iACLs to protect the control plane of all network devices.

See the earlier section on iACLs in this document for more information.

CoPP

The CoPP feature can also be used to restrict IP packets that are destined for the infrastructure device itself and require control-plane CPU processing. CoPP in Cisco NX-OS can be used to police different classes of traffic to different permitted levels, effectively applying quality of service (QoS) to control-plane-bound traffic.

The configuration of CoPP is similar to data-plane QoS configuration and uses the same Modular QoS CLI (MQC) configuration structures:

- Class maps are defined to match specific types of traffic
- Policy maps are created to apply policing (rate-limiting) policies to class-map-matched traffic
- A service policy is used to map the policy map to the control-plane interface

Cisco NX-OS provides simplified setup for typical network environments by offering predefined class maps and policy maps using the initial configuration setup script. When you run the setup script, or at bootup, you can select one of four predefined templates to be applied for CoPP:

- Strict
- Moderate
- Loose
- None

After a CoPP template is selected, it will be applied to the control-plane interface. If the CoPP policy is changed from one of the actively policing templates (strict, moderate, or loose) to none, the system will not remove the existing class maps or policy maps. It will simple not map the policy map to the control-plane interface with a service policy. This approach leaves the configuration in place but simply does not apply it to the interface. If a different active policing template is chosen to replace one in place, the template will overwrite the existing class maps and policy maps with the new settings.

In this example, a CoPP configuration is created in which SSH traffic only from trusted hosts is permitted to reach the Cisco NX-OS device CPU. All other control-plane traffic is allowed:

Note: Dropping traffic from unknown or untrusted IP addresses can prevent hosts with dynamically assigned IP addresses from connecting to the Cisco NX-OS device.

```
!
access-list ALLOW_TRUSTED_SSH
   deny tcp <trusted-addresses> <mask> any eq 22
   permit tcp any any eq 22
   deny ip any any
!
class-map type control-plane match-all COPP-KNOWN-UNDESIRABLE
   match access-group name ALLOW_TRUSTED_SSH
!
policy-map type control-plane COPP-INPUT-POLICY
   class COPP-KNOWN-UNDESIRABLE
          police 1 conform drop violate drop
!
control-plane
   service-policy input COPP-INPUT-POLICY
!
```

In the preceding CoPP example, the ACL entries that match the unauthorized packets with the **permit** action result in a discard of these packets by the policy-map **drop** function, while packets that match the **deny** action are not affected by the policy-map **drop** function.

This example illustrates the theory, structure, and applicability of CoPP. In reality, an effective CoPP policy is more complex than the simplified example shown here and requires adequate planning and testing before being deployed in a live production environment.

Refer to the "Configuring Control-Plane Policing" section of the Cisco NX-OS Security Configuration Guide for more information about the configuration and use of the CoPP feature.

Securing the Data Plane

Although the data plane is responsible for moving data from the source to the destination, within the context of security the data plane is the least important of the three planes. For this reason, when securing a network device you should protect the management and control planes in preference over the data plane. However, within the data plane itself, there are many features and configuration options that can help secure traffic. The following sections detail these features and options so that you can more easily secure your network.

General Data-Plane Hardening

Most data plane traffic flows across the network as determined by the network's routing configuration. However, IP network functions are available to alter the path of packets across the network. Features such as IP options - specifically, the source routing option - create security challenges in today's networks.

The use of tACLs is also relevant to the hardening of the data plane. See the "Filtering Transit Traffic with tACLs" section of this document for more information.

Disabling IP Source Routing

IP source routing uses the Loose Source Route and Record Route options in tandem or the Strict Source Route along with the Record Route option to enable the source of the IP datagram to specify the network path that a packet takes. This function can be used in attempts to route traffic around security controls in the network.

If IP options have not been completely disabled though the IP Options Selective Drop feature, it is important that you disable IP source routing. IP source routing, which is enabled by default in all Cisco NX-OS releases, is disabled through the **no ip source-route** global configuration command. This configuration example illustrates the use of this command:

```
!
no ip source-route
!
```

Refer to Cisco NX-OS Command Reference for more information about the **ip source-route** command.

Disabling ICMP Redirect Messages

ICMP redirect messages are used to inform a network device of a better path to an IP destination. By default, Cisco NX-OS sends a redirect message if it receives a packet that must be routed through the interface from which it was received.

In some situations, an attacker may be able to cause the Cisco NX-OS device to send many ICMP redirect messages, resulting in an elevated CPU load. For this reason, the transmission of ICMP redirect messages should be disabled. ICMP redirect messages are disabled using the interface configuration command **no ip redirects**, as shown in the example configuration:

```
!
interface ethernet 1/1
          no ip redirects
!
```

Refer to the Cisco NX-OS Command Reference for more information about the **ip redirects** interface configuration command.

Disabling or Limiting IP Directed Broadcasts

IP directed broadcasts make it possible to send an IP broadcast packet to a remote IP subnet. After the packet reaches the remote network, the forwarding IP device sends the packet as a Layer 2 broadcast to all stations on the subnet. This directed broadcast function has been used as an amplification and reflection aid in several attacks, including the smurf attack.

Current versions of Cisco NX-OS have this function disabled by default; however, it can be enabled with the **ip directed-broadcast** interface configuration command.

Refer to "Configuring IPv4" in the Cisco NX-OS Unicast Routing Configuration Guide for more information about the **ip directed-broadcast** command.

Filtering Transit Traffic with tACLs

You can control what traffic transits the network by using tACLs. In contrast, iACLs seek to filter traffic that is destined for the network itself. The filtering provided by tACLs is beneficial when it is desirable to filter traffic to a particular group of devices or traffic that is transiting the network.

This type of filtering is traditionally performed by firewalls. However, in some instances it may be beneficial to perform this filtering on a Cisco NX-OS device in the network: for example, when filtering must be performed but no firewall is present.

A tACLs is also an appropriate place in which to implement static antispoofing protections. See the "Implementing Antispoofing Protection" section of this document for more information.

Refer to the document "Transit Access Control Lists: Filtering at Your Edge" for more information about tACLs.

Filtering ICMP Packets

ICMP was designed as a control protocol for IP. As such, the messages it conveys can have far-reaching ramifications on the TCP and IP protocols in general. ICMP is used by the network troubleshooting tools **ping** and **traceroute**, as well as by path MTU discovery; however, external ICMP connectivity is rarely needed for the proper operation of a network.

Cisco NX-OS provides functions to specifically filter ICMP messages by name or type and code. This example ACL allows ICMP from trusted networks while blocking all ICMP packets from other sources:

```
!
ip access-list ACL-TRANSIT-IN
! !--- Permit ICMP packets from trusted networks only !
 permit icmp <trusted-networks>/<mask> any
! !--- Deny all other IP traffic to any network device !
 deny icmp any any
!
```

Filtering IP Fragments

As discussed previously in the "Limiting Access to the Network with Infrastructure ACLs" section of this document, the filtering of fragmented IP packets can pose a challenge to security devices.

Because of the non-intuitive nature of fragment handling, IP fragments are often inadvertently permitted by ACLs. Fragmentation is also often used in attempts to evade detection by intrusion-detection systems. For these reasons, IP fragments are often used in attacks and should be explicitly filtered at the top of any configured tACLs. The ACL shown here includes comprehensive filtering of IP fragments. The function illustrated in this example must be used in conjunction with the functions shown in the previous examples:

```
!
ip access-list ACL-TRANSIT-IN
!
!--- Deny IP fragments using protocol-specific ACEs to aid in
!--- classification of attack traffic !
   deny tcp any any fragments
   deny udp any any fragments
   deny icmp any any fragments
   deny ip any any fragments
!
```

Implementing Antispoofing Protection

Many attacks use source IP address spoofing to be effective or to conceal the true source of an attack and hinder accurate traceback. Cisco NX-OS provides uRPF and IP source guard to deter attacks that rely on source IP address spoofing. In addition, ACLs and null routing are often deployed as manual means of spoofing prevention.

IP source guard is effective at reducing spoofing for networks that are under direct administrative control by performing switch port, MAC address, and source address verification. uRPF provides source network verification and can reduce spoofed attacks from networks that are not under direct administrative control. Port security can be used to validate MAC addresses at the access layer. Dynamic ARP Inspection (DAI) mitigates attack vectors that use ARP poisoning on local segments.

Configuring uRPF

uRPF enables a device to verify that the source address of a forwarded packet can be reached through the interface that received the packet. You must not rely on uRPF as the only protection against spoofing. Spoofed packets can enter the network through a uRPF-enabled interface if an appropriate return route to the source IP address exists. uRPF relies on you to enable Cisco Express Forwarding on each device, and it is configured on a per-interface basis.

uRPF can be configured in either of two modes: loose or strict. In cases in which asymmetric routing exists, loose mode is preferred because strict mode is known to drop packets in these situations. During configuration of the **ip verify** interface configuration command, the keyword any configures loose mode, and the keyword **any** configures strict mode.

This example illustrates configuration of this feature:

```
!
interface Ethernet <slot>/<port>
   ip verify unicast source reachable-via [any | rx]
!
```

Refer to the section “Configuring Unicast RPF” in the Cisco NX-OS Security Configuration Guide for more information about the configuration and use of uRPF.

Using IP Source Guard

IP source guard is an effective means of spoofing prevention that can be used if you have control over Layer 2 interfaces. IP source guard uses information from Dynamic Host Configuration Protocol (DHCP) snooping to dynamically configure a port ACL (PACL) on the Layer 2 interface, denying any traffic from IP addresses that are not associated in the IP source binding table.

IP source guard can be applied to Layer 2 interfaces belonging to VLANs enabled for DHCP snooping. These commands enable DHCP snooping:

```
!
feature dhcp
ip dhcp snooping
!
```

For more information about DHCP snooping and the various configuration options, please refer to the "Configuring DHCP Snooping" section of the Cisco NX-OS Security Configuration Guide.

After DHCP snooping is enabled, these commands enable IP source guard:

```
!
interface ethernet <slot>/<port>
   ip verify source dhcp-snooping vlan
!
```

Refer to "Configuring Source Guard" in the Cisco NX-OS Security Configuration Guide for more information about this feature.

Using Port Security

Port security is used to mitigate MAC address spoofing at the access interface. Port security can use dynamically learned (sticky) MAC addresses to facilitate the initial configuration. After port security has determined a MAC address violation, it can use one of four violation modes: protect, restrict, shutdown, and shutdown VLAN. In instances in which a port provides access only for a single workstation using standard protocols, a maximum value of 1 may be sufficient. Protocols that using virtual MAC addresses such as Hot Standby Router Protocol (HSRP) do not function when the maximum value is set to 1.

```
!
feature port-security
interface <slot>/<port>
   switchport
   switchport port-security [mac address sticky]
!-- Optionally enable sticky MAC address learning
!
```

Refer to "Configuring Port Security" in the Cisco NX-OS Security Configuration Guide for more information about configuring port security.

Using DAI

DAI can be used to mitigate ARP poisoning attacks on local segments. An ARP poisoning attack is a method in which an attacker sends falsified ARP information to a local segment. This information is designed to corrupt the ARP cache of other devices. Often an attacker uses ARP poisoning to perform a man-in-the-middle attack.

DAI intercepts and validates the IP-to-MAC address relationship of all ARP packets on untrusted ports. In DHCP environments, DAI uses the data that is generated by the DHCP snooping feature. ARP packets that are received on trusted interfaces are not validated, and invalid packets on untrusted interfaces are discarded. In non-DHCP environments, the use of ARP ACLs is required.

These commands enable DHCP snooping:

```
!
feature dhcp
ip dhcp snooping
ip dhcp snooping vlan <vlan-range>
!
```

After DHCP snooping has been enabled, these commands enable DAI:

```
!
ip arp inspection vlan <vlan-range>
!
```

In non-DHCP environments, ARP ACLs are required to enable DAI. This example demonstrates the basic configuration of DAI with ARP ACLs:

```
!
arp access-list <acl-name> permit ip host <sender-ip> mac host <sender-mac>
!
ip arp inspection filter <arp-acl-name> vlan <vlan-range>
!
```

Refer to "Configuring Dynamic ARP Inspection" in the Cisco NX-OS Security Configuration Guide for more information about how to configure DAI.

Configuring Antispoofing ACLs

Manually configured ACLs can also provide static antispoofing protection against attacks that use known unused and untrusted address space. Commonly, these antispoofing ACLs are applied to ingress traffic at network boundaries as a component of a larger ACL. Antispoofing ACLs require regular monitoring because they can change frequently. Spoofing can be reduced in traffic originating from the local network by applying outbound ACLs that limit the traffic to valid local addresses.

This example demonstrates how ACLs can be used to limit IP spoofing. This ACL is applied inbound on the desired interface. The access control entries that make up this ACL are not comprehensive. If you configure these types of ACLs, seek an up-to-date reference that is conclusive.

```
!
ip access-list ACL-ANTISPOOF-IN
   deny ip 10.0.0.0 0.255.255.255 any
   deny ip 192.168.0.0 0.0.255.255 any
!
interface ethernet <slot>/<port>
   ip access-group ACL-ANTISPOOF-IN in
!
```

Refer to the Cisco whitepaper "Configuring Commonly Used IP ACLs" for more information about how to configure static antispoofing ACLs.

The official list of unallocated Internet addresses is maintained by Team Cymru[1]. Additional information about filtering unused addresses is available at the Bogon reference page[2].

Limiting the Effect of Data-Plane Traffic on the CPU

The primary purpose of routers and switches is to forward packets and frames through the device onward to final destinations. These packets, which transit the devices deployed throughout the network, can affect the CPU operations of a device. The data plane, which consists of traffic transiting the network device, should be secured to help ensure the operation of the management and control planes. If transit traffic can cause a device to process switch traffic, the control plane of a device can be affected, which may lead to disruption of operations.

Features and Traffic Types That Affect the CPU

Although not exhaustive, this list includes types of data plane traffic that may require special CPU processing and are process switched by the CPU:

- ACL logging: ACL logging traffic consists of any packets that are generated due to a match (permit or deny) of an access control entry on which the **log** keyword is used.
- uRPF: uRPF used in conjunction with an ACL may result in the process switching of certain packets.
- IP options: Any IP packets with options included must be processed by the CPU.
- Fragmentation: Any IP packet that requires fragmentation must be passed to the CPU for processing.
- TTL expiry: Packets that have a TTL value less than or equal to 1 require ICMP Time Exceeded (ICMP Type 11, Code 0) messages to be sent, which results in CPU processing.
- ICMP unreachable messages: Packets that result in ICMP unreachable messages due to routing, MTU, or filtering are processed by the CPU.
- Traffic requiring an ARP request: Destinations for which an ARP entry does not exist require processing by the CPU.
- Non-IP traffic: All non-IP traffic is processed by the CPU.

CPU handling of special data-plane packets is platform dependent. The architecture of the specific Cisco NX-OS platform will dictate what can and cannot be processed by hardware and what must be passed to the CPU.

Refer to the platform-specific hardware implementation details for a given device to determine what types of data-plane traffic may affect the system CPU.

See the "General Data-Plane Hardening" section of this document for more information about securing the data plane.

[1] http://www.team-cymru.org/Services/Bogons/bogon-dd.html
[2] http://www.team-cymru.org/Services/Bogons/

Traffic Identification and Traceback

At times, you may need to quickly identify and trace back network traffic, especially during incident response or poor network performance. NetFlow and classification ACLs are the two primary mechanisms for accomplishing this using Cisco NX-OS. NetFlow can provide visibility into all traffic on the network. Additionally, NetFlow can be implemented with collectors that can provide long-term trending and automated analysis. Classification ACLs are a component of ACLs and require planning to identify specific traffic and manual intervention during analysis. These sections provide a brief overview of each feature.

NetFlow

NetFlow identifies anomalous and security-related network activity by tracking network flows. NetFlow data can be viewed and analyzed using the CLI, or the data can be exported to a commercial or freeware NetFlow collector for aggregation and analysis. NetFlow collectors, through long-term trending, can provide network behavior and usage analysis. NetFlow functions by performing analysis on specific attributes within IP packets and creating flows. NetFlow Version 5 is the most commonly used version of NetFlow; however, Version 9 is more extensible. NetFlow flows can be created using sampled traffic data in high-volume environments.

This example illustrates the basic configuration of this feature:

```
!
! - Enable the Netflow feature
feature netflow
!
! - Define a flow record.  There are also predefined standard records that can be
used.
flow record FLOW_RECORD_EXAMPLE
   description Example flow record
   match ip protocol
   collect counter bytes
   collect flow direction
   collect interface input
   collect interface output
   collect timestamp sys-uptime first
   collect timestamp sys-uptime last
!
! - Create a flow exporter
flow exporter EXAMPLE_FLOW_EXPORTER
   destination <IP address> use-vrf <vrf name>
   source <interface>
   version {5 | 9}
!
! - Create a flow monitor
flow monitor EXAMPLE_FLOW_MONITOR
   description <string>
   exporter EXAMPLE_FLOW_EXPORTER
   record { EXAMPLE_FLOW_RECORD | netflow-original | netflow protocol-port }
!
```

```
! - Apply the flow monitor to an interface
interface ethernet <slot>/<port>
   ip flow monitor EXAMPLE_FLOW_MONITOR {input | output}
!
```

For more information about configuration of Netflow in Cisco NX-OS, refer to the "Configuring NetFlow" section of the Cisco NX-OS System Management Configuration Guide.

Refer to the Cisco whitepaper, "Introduction to Cisco IOS NetFlow: A Technical Overview" for a general technical overview of NetFlow.

Classification ACLs

Classification ACLs provide visibility into traffic that traverses an interface. Classification ACLs do not alter the security policy of a network and are typically constructed to classify individual protocols, source addresses, or destinations. For example, an access control entry that permits all traffic could be separated into specific protocols or ports. This more detailed classification of traffic into specific access control entries can help provide an understanding of the network traffic because each traffic category has its own hit counter. An administrator can also separate the implicit deny response at the end of an ACL into granular access control entries to help identify the types of denied traffic.

An administrator can expedite an incident response by using classification ACLs with the **show access-list** and **clear ip access-list counters** EXEC commands.

This example illustrates the configuration of a classification ACL to identify small and medium-sized business (SMB) traffic prior to a default deny response:

```
!
ip access-list ACL-SMB-CLASSIFY
remark Existing contents of ACL
remark Classification of SMB specific TCP traffic
deny tcp any any eq 139
deny tcp any any eq 445
deny ip any any
!
```

To identify traffic that uses a classification ACL, use the **show access-list acl-name** EXEC command. The ACL counters can be cleared by using the **clear ip access-list counters acl-name** EXEC command.

Access Control with VLAN Maps and PACLs

VLAN ACLS (VACLs), or VLAN maps and PACLs, provide the capability to enforce access control on nonrouted traffic that is closer to endpoint devices than ACLs that are applied to routed interfaces.

The following sections provide an overview of the features, benefits, and potential Cisco NX-OS use cases for VACLs and PACLs.

Access Control with VLAN Maps

VACLs, or VLAN maps that apply to all packets that enter the VLAN, provide the capability to enforce access control for intra-VLAN traffic. This control is not possible using ACLs on routed interfaces. For example, a VLAN map can be used to prevent hosts that are contained within the same VLAN from communicating with each other, thereby reducing opportunities for local attackers or worms to exploit a host on the same network segment. To prevent packets from using a VLAN map, you can create an ACL that matches the traffic and, in the VLAN map, set the action to **drop**. After a VLAN map is configured, all packets that enter the LAN are sequentially evaluated against the configured VLAN map. VLAN access maps support IPv4 and MAC address access lists; however, they do not support logging or IPv6 ACLs.

This example uses an extended named access list to illustrate the configuration of this feature:

```
!
ip access-list <acl-name>
permit <protocol> <src-address> <src-port> <dst-address> <dst-port>
!
vlan access-map <name> <number>
   match ip address <acl-name>
   action <drop|forward>
!
```

This example demonstrates the use of a VLAN map to deny access to TCP ports 139 and 445:

```
!
ip access-list VACL-MATCH-ANY
   permit ip any any
!
ip access-list VACL-MATCH-PORTS
   permit tcp 192.168.1.0 0.0.0.255 192.168.1.0 0.0.0.255 eq 445
   permit tcp 192.168.1.0 0.0.0.255 192.168.1.0 0.0.0.255 eq 139
!
!
vlan access-map VACL 20
   match ip address VACL-MATCH-PORTS
   action drop
!
vlan access-map VACL 30
   match ip address VACL-MATCH-ANY
   action forward
!
```

Refer to the "Configuring Network Security with ACLs" section of the Catalyst Switch Software Configuration Guide for general information about the configuration of VLAN maps.

Refer to the "Configuring VLAN ACLs" section of the Cisco NX-OS Security Configuration Guide for more information about configuring VLAN maps in Cisco NX-OS.

Access Control with PACLs

PACLs can be applied only to the inbound direction on Layer 2 physical interfaces of a switch. Similar to VLAN maps, PACLs provide access control on unrouted or Layer 2 traffic. The syntax for creating PACLs, which take precedence over VLAN maps and router ACLs, is the same as for router ACLs. If an ACL is applied to a Layer 2 interface, then it is referred to as a PACL. Configuration involves creating an IPv4, IPv6, or MAC address ACL and applying it to the Layer 2 interface.

This example uses an extended named access list to illustrate the configuration of this feature:

```
!
ip access-list extended <acl-name> permit <protocol> <source-address>
<source-port> <destination-address> <destination-port>
!
interface <type> <slot/port> switchport mode access switchport access vlan
<vlan_number> ip access-group <acl-name> in
!
```

Refer to the "Port ACLs" section of the Catalyst Switch Software Configuration Guide - Configuring Network Security with ACLs for more information about the configuration of PACLs.

Access Control with MAC Address ACLs

MAC packet classification allows you to control whether a MAC ACL that is on a Layer 2 interface applies to all traffic entering the interface, including IP traffic, or to non-IP traffic only.

You can enable or disable MAC packet classification only on Layer 2 interfaces. To configure an interface as Layer 2, use the 'switchport' command.

The example below shows how to configure an Ethernet interface to operate as Layer 2 and how to enable MAC packet classification:

```
switch# conf t
switch(config)# interface ethernet 2/3
switch(config-if)# switchport
switch(config-if)# mac packet-classify
switch(config-if)#
```

Note: The command is supported in the Cisco NX-OS Software Release 4.2(1) or later releases.

This interface command must be applied on the ingress interface, and it instructs the forwarding engine to not inspect the IP header. As a result, you can use a MAC access list on IP traffic.

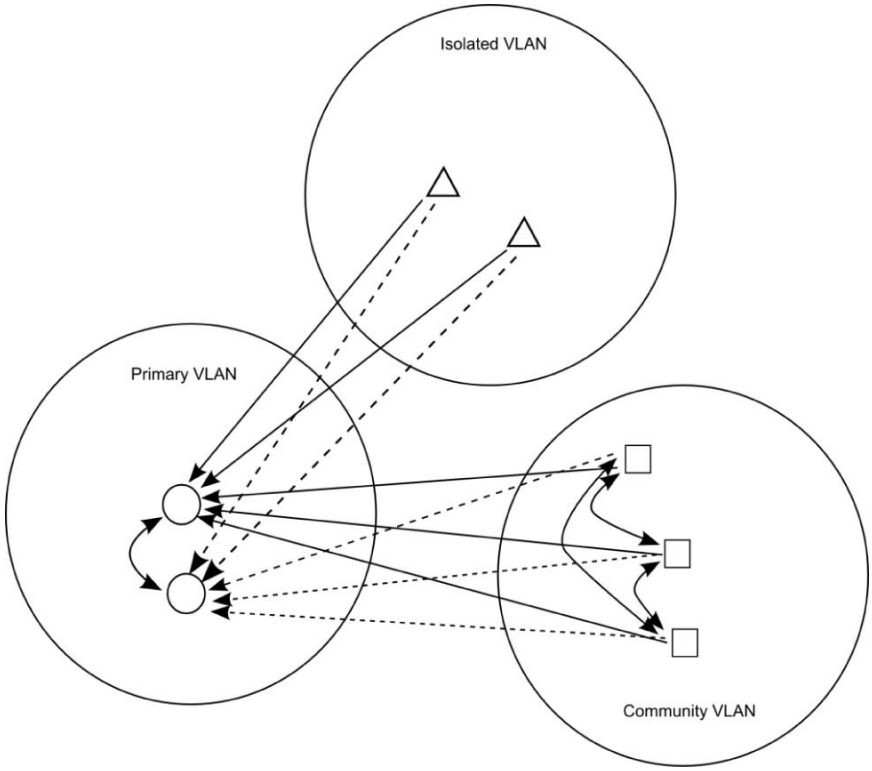
Private VLANs

Private VLANs (PVLANs) are a Layer 2 feature that limits connectivity between workstations or servers within a VLAN. Without PVLANs, all devices on a Layer 2 VLAN can communicate freely. In some networking situations, security can be aided by limiting communication between devices on a single VLAN. Limiting the communications patterns possible on a VLAN by using PVLANs can provide an effective security tool. For example, PVLANs are often used to prohibit communication between servers in a publicly accessible subnet.

If a single server becomes compromised, the lack of connectivity to other servers due to the application of PVLANs can help limit the compromise to the one server.

There are three types of VLAN constructs in the context of PVLANs: isolated VLANs, community VLANs, and primary VLANs. The configuration of PVLANs makes use of primary and secondary VLANs. The primary VLAN contains all promiscuous ports, which are mapped for one-to-many relationships to nodes on other VLAN types, which include one or more secondary VLANs that can be either isolated or community VLANs (Figure 1).

Figure 1. Relationship Between VLAN Types and Ports in PVLANs



Isolated VLANs

The configuration of a secondary VLAN as an isolated VLAN completely prevents communication between devices in the secondary VLAN. There can be only one isolated VLAN per primary VLAN, and only promiscuous ports can communicate with ports in an isolated VLAN. Isolated VLANs should be used on untrusted networks and in situations in which there is no trust relationship between nodes, such as on networks that support guests.

This example configures VLAN 11 as an isolated VLAN and associates it with the primary VLAN, VLAN 20. The example also configures interface Ethernet 1/1 as an isolated port in VLAN 11:

```
! - In order to use Private VLANs the feature must first be enabled
feature private-vlan
!
vlan 11 private-vlan isolated
!
vlan 20
   private-vlan primary
   private-vlan association 11
!
interface ethernet 1/1
```

```
    description *** Port in Isolated VLAN ***
    switchport mode private-vlan host
    switchport private-vlan host-association 20 11
!
```

Community VLANs

A secondary VLAN that is configured as a community VLAN allows communication among members of the VLAN as well as with any promiscuous ports in the primary VLAN. However, no communication is possible between any two community VLANs or from a community VLAN to an isolated VLAN. Community VLANs must be used to group servers that need connectivity to one another, but for which connectivity to all other devices in the VLAN is not required. This scenario is common in a publicly accessible network or anywhere that servers provide content to untrusted clients but must maintain an internal trust and relationship between themselves for normal operation.

This example configures a single community VLAN and configures switch port Ethernet 1/2 as a member of that VLAN. The community VLAN, VLAN 12, is a secondary VLAN to primary VLAN 20.

```
!
vlan 12
    private-vlan community
!
vlan 20
    private-vlan primary
    private-vlan association 12
!
interface ethernet 1/2
    description *** Port in Community VLAN ***
    switchport mode private-vlan host
    switchport private-vlan host-association 20 12
!
```

Promiscuous Ports

Switch ports that are placed in the primary VLAN are known as promiscuous ports. Promiscuous ports can communicate with all other ports in the primary and secondary VLANs. Router or firewall interfaces are the most common devices found on these VLANs.

This configuration example combines the previous isolated and community VLAN examples and adds the configuration of interface Ethernet 1/12 as a promiscuous port:

```
!
feature private-vlan
!
vlan 11
    private-vlan isolated
!
vlan 12
    private-vlan community
!
vlan 20
```

```
    private-vlan primary
    private-vlan association 11-12
!
interface ethernet 1/1
    description *** Port in Isolated VLAN ***
    switchport mode private-vlan host
    switchport private-vlan host-association 20 11
!
interface ethernet 1/2
    description *** Port in Community VLAN ***
    switchport mode private-vlan host
    switchport private-vlan host-association 20 12
!
interface ethernet 1/12
    description *** Promiscuous Port ***
    switchport mode private-vlan promiscuous
    switchport private-vlan mapping 20 add 11-12
!
```

When implementing PVLANs, you must help ensure that the Layer 3 configuration in place supports the restrictions that are imposed by PVLANs and does not allow the PVLAN configuration to be subverted by routing. Layer 3 filtering using a router ACL or firewall can prevent the subversion of the PVLAN configuration.

Refer to Private VLANs (PVLANs): Promiscuous, Isolated, Community, homepage for more information about the use of PVLANs as a security tool.

Refer to “Configuring Private VLANs Using Cisco NX-OS” in the Cisco NX-OS Layer 2 Switching Configuration Guide for more information about configuring PVLANs in Cisco NX-OS.

Conclusion

This document provides a broad overview of the methods that can be used to secure a Cisco NX-OS system device. By securing the individual devices, you increase the overall security of the networks that you manage. In this overview, protection of the management, control, and data planes is discussed, and recommendations for configuration are supplied. Where possible, sufficient detail is provided for the configuration of each associated feature. However, in all cases, comprehensive references are provided to supply you with the information needed for further evaluation.

Appendix A: Cisco NX-OS Hardening Checklist

Management Plane

- Enable strong password checking
- Use an encrypted transport protocol for CLI access (SSH)
- Use AAA for authentication
- Use AAA (TACACS+) for command authorization
- Use AAA for accounting
- Apply an ACL to vty access
- Set the EXEC timeout appropriately
- Apply iACLs to all interfaces facing untrusted noninfrastructure devices
- Set up ICMP filtering as needed
- Adjust ACLs to properly filter for IP fragments
- Develop and configure a warning or message-of-the-day (MOTD) banner
- Configure redundant AAA servers
- Configure AAA fallback to local (if applicable)
- Configure SNMPv3 (if applicable)
- Configure SNMPv2 communities and apply ACLs (if applicable)
- Configure centralized logging
- Set logging levels for all relevant components
- Set the logging source interface
- Configure logging time-stamp granularity

Control Plane

- Disable ICMP redirect messages
- Disable ICMP unreachable messages as needed
- Configure NTP authentication if NTP is being used
- Configure CoPP
- Disable IP source routing
- Disable IP directed broadcasts

Data Plane

- Configure tACLs
- Configure required antispoofing protections (ACLs, IP source guard, DAI, uRPF, etc.)
- Configure NetFlow (if needed)
- Configure classification ACLs (if needed)
- Configure required ACLs (if needed)
- Configure Private VLANs (if needed)

Appendix B: Enabling FIPS Mode

Cisco NX-OS supports FIPS mode to meet the requirements of FIPS 140-2. The requirements outlined in the FIPS140-2 publication "Security Requirements for Cryptographic Modules" specify certain characteristics that must be met in the cryptographic modules and components of a platform for the platform to be considered secure.

Enabling FIPS mode in Cisco NX-OS does the following:

- Enables FIPS self-tests that are performed at boot time
- Enables the FIPS error state if the FIPS self-test fails at boot time
- Enables RADIUS key wrap

FIPS mode is typically required in U.S. Department of Defense (DoD) and other U.S. government deployments. It can be used in common enterprise or service provider environments but is not required for strong security in those environments. The decision to enable FIPS mode or not is environment specific and requires internal security policy analysis and planning.

FIPS mode may not be available in export versions of Cisco NX-OS in some countries due to export regulations.

For more information, refer to the "Configuring FIPS" section of the Cisco NX-OS Security Configuration Guide.

For further information, questions and comments please contact ccbu-pricing@cisco.com

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA C07-665160-02 04/14