Parka document for CCNA Security 640-554 (which retires 11/30/2015)

IPS Inline Mode Event Actions

The following actions require the device to be deployed in Inline mode.

Deny attacker inline: This action is the most severe and effectively blocks all communication from the attacking host that passes through the IPS for a specified period of time. Because this event action is severe, administrators are advised to use this only when the probability of false alarms or spoofing is minimal.

Deny attacker service pair inline: This action prevents communication between the attacker IP address and the protected network on the port in which the event was detected. However, the attacker would be able to communicate on another port that has hosts on the protected network. This event action works well for worms that attack many hosts on the same service port. If an attack occurred on the same host but on another port, this communication would be allowed. This event action is appropriate when the likelihood of a false alarm or spoofing is minimal.

Deny attacker victim pair inline: This action prevents the attacker from communicating with the victim on any port. However, the attacker could communicate with other hosts, making this action better suited for exploits that target a specific host. This event action is appropriate when the likelihood of a false alarm or spoofing is minimal.

Deny connection inline: This action prevents further communication for the specific TCP flow. This action is appropriate when there is the potential for a false alarm or spoofing and when an administrator wants to prevent the action but not deny further communication.

Deny packet inline: This action prevents the specific offending packet from reaching its intended destination. Other communication between the attacker and victim or victim network may still exist. This action is appropriate when there is the potential for a false alarm or spoofing. Note that for this action, the default time has no effect.

Modify packet inline: This action enables the IPS device to modify the offending part of the packet. However, it forwards the modified packet to the destination. This action is appropriate for packet normalization and other anomalies, such as TCP segmentation and IP fragmentation re-ordering.

Atomic Engine in IPS

The Atomic engine contains **signatures for simple, single packet conditions that cause alerts to be fired**.

Monitoring Cisco IOS IPS Signatures via Syslog or SDEE

Cisco IOS IPS provides two methods to report IPS intrusion alerts—Cisco IOS logging (syslog) and Security Device Event Exchange (SDEE). Use this task to enable SDEE to report IPS intrusion alerts.

SDEE Overview

SDEE is an **secure application-level communication protocol that is used to exchange IPS messages between IPS clients and IPS servers**.

Prerequisites

To use SDEE, the HTTP server must be enabled (via the **ip http server** command). If the HTTP server is not enabled, the router cannot respond to the SDEE clients because it cannot not see the requests.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ips notify sdee**
4. **ip sdee events** *events*
5. **ip sdee subscriptions** *subscriptions*
6. **exit**
7. **show ip sdee** {[**alerts**] [**all**] [**errors**] [**events**] [**configuration**] [**status**] [**subscriptions**]}

Understanding Anomaly Detection

Anomaly detection assumes it gets traffic from both directions. If the sensor is configured to see only one direction of traffic, you should turn off anomaly detection. Otherwise, when anomaly detection is running in an asymmetric environment, it identifies all traffic as having incomplete connections, that is, as scanners, and sends alerts for all traffic flows. Using asymmetric mode protection with anomaly detection enabled causes excessive resource usage and possible false positives for anomaly detection signatures.

The anomaly detection component of the sensor detects worm-infected hosts. This enables the sensor to be less dependent on signature updates for protection again worms and scanners, such as Code Red and SQL Slammer and so forth. The anomaly detection component lets the sensor learn normal activity and send alerts or take dynamic response actions for behavior that deviates from what it has learned as normal behavior.

Anomaly detection detects the following two situations:

- When the network starts on the path of becoming congested by worm traffic.
- When a single worm-infected source enters the network and starts scanning for other vulnerable hosts.

Worms

Anomaly detection assumes it gets traffic from both directions. If the sensor is configured to see only one direction of traffic, you should turn off anomaly detection. Otherwise, when anomaly detection is running in an asymmetric environment, it identifies all traffic as having incomplete connections, that is, as scanners, and sends alerts for all traffic flows. Using asymmetric mode protection with anomaly detection enabled causes excessive resource usage and possible false positives for anomaly detection signatures.

Worms are automated, self-propagating, intrusion agents that make copies of themselves and then facilitate their spread. Worms attack a vulnerable host, infect it, and then use it as a base to attack other vulnerable hosts. They search for other hosts by using a form of network inspection, typically a scan, and then propagate to the next target. A scanning worm locates vulnerable hosts by generating a list of IP addresses to probe, and then contacts the hosts. Code Red worm, Sasser worm, Blaster worm, and the Slammer worm are examples of worms that spread in this manner.

Anomaly detection identifies worm-infected hosts by their behavior as scanners. To spread, a worm must find new hosts. It finds them by scanning the Internet or network using TCP, UDP, and other protocols to generate unsuccessful attempts to access different destination IP addresses. A scanner is defined as a source IP address that generates events on the same destination port (in TCP and UDP) for too many destination IP addresses.

The events that are important for TCP protocol are nonestablished connections, such as a SYN packet that does not have its SYN-ACK response for a given amount of time. A worm-infected host that scans using TCP protocol generates nonestablished connections on the same destination port for an anomalous number of IP addresses.

The events that are important for UDP protocol are unidirectional connections, such as a UDP connection where all packets are going only in one direction. A worm-infected host that scans using UDP protocol generates UDP packets but does not receive UDP packets on the same quad within a timeout period on the same destination port for multiple destination IP addresses.

The events that are important for other protocols, such as ICMP, are from a source IP address to many different destination IP addresses, that is, packets that are received in only one direction.

If a worm has a list of IP addresses it should infect and does not have to use scanning to spread itself (for example, it uses passive mapping—listening to the network as opposed to active scanning), it is not detected by the anomaly detection worm policies. Worms that receive a mailing list from probing files within the infected host and email this list are also not detected, because no Layer 3/Layer 4 anomaly is generated.

WEB Security Appliance (from the marketing PDF)

Benefits • Get advanced threat detection through integration with Cisco Advanced Malware Protection, Cisco Cognitive Threat Analytics, and cloud access security products. • Easily deploy the solution with fast, flexible options and automatic updates. • Enhance security with highly detailed web-access control with the industry-leading Cisco Identity Services Engine. • Reduce costs with easy integration into your existing security infrastructure. • Get an all-in-one **solution with web security** and proxy capabilities supported within a single box.

The Phishing Problem

Phishing is an attempt to fraudulently acquire sensitive information (such as usernames, passwords, and credit card details) by masquerading as a trustworthy entity in an electronic communication. Phishing is typically carried out by email and often directs users to enter details at a fake website.

The individuals behind phishing send out millions of emails in the hope that a few recipients will act on them. Any email address that has been made public (in forums, in newsgroups, or on a website) is susceptible to phishing.

Information About Unicast RPF

The **Unicast RPF feature reduces problems that are caused by the introduction of malformed or forged (spoofed) IPv4 or IPv6 source addresses** into a network by discarding IPv4 or IPv6 packets that lack a verifiable IP source address. For example, a number of common types of Denial-of-Service (DoS) attacks, including Smurf and Tribal Flood Network (TFN) attacks, can take advantage of forged or rapidly changing source IPv4 or IPv6 addresses to allow attackers to thwart efforts to locate or filter the attacks. Unicast RPF deflects attacks by forwarding only the packets that have source addresses that are valid and consistent with the IP routing table.

When you enable Unicast RPF on an interface, the device examines all ingress packets received on that interface to ensure that the source address and source interface appear in the routing table and match the interface on which the packet was received. This examination of source addresses relies on the Forwarding Information Base (FIB).

Unicast RPF verifies that any packet received at a device interface arrives on the best return path (return route) to the source of the packet by doing a reverse lookup in the FIB. If the packet was received from one of the best reverse path routes, the packet is forwarded as normal. If there is no reverse path route on the same interface from which the packet was received, the source address might have been modified by the attacker. If Unicast RPF does not find a reverse path for the packet, the packet is dropped.

Note With Unicast RPF, all equal-cost "best" return paths are considered valid, which means that Unicast RPF works where multiple return paths exist, if each path is equal to the others in terms of the routing cost (number of hops, weights, and so on) and as long as the route is in the FIB. Unicast RPF also functions where Enhanced Interior Gateway Routing Protocol (EIGRP) variants are being used and unequal candidate paths back to the source IP address exist.

Unicast RPF Process

Unicast RPF has several key implementation principles:

- **The packet must be received at an interface that has the best return path (route) to the packet source (a process called *symmetric routing*).** There must be a route in the FIB that matches the route to the receiving interface. Static routes, network statements, and dynamic routing add routes to the FIB.
- IP source addresses at the receiving interface must match the routing entry for the interface.
- Unicast RPF is an input function and is applied only on the input interface of a device at the upstream end of a connection.

You can use Unicast RPF for downstream networks, even if the downstream network has other connections to the Internet.

How NAT is implemented on the ASA

The ASA can implement address translation in two ways: ***network object NAT* and *twice NAT*** .

Main Differences Between Network Object NAT and Twice NAT

The main differences between these two NAT types are:

- How you define the real address.
  - Network object NAT—You define NAT as a parameter for a network object. A network object names an IP host, range, or subnet so you can then use the object in configuration instead of the actual IP addresses. The network object IP address serves as the real address. This method lets you easily add NAT to network objects that might already be used in other parts of your configuration.
  - Twice NAT—You identify a network object or network object group for both the real and mapped addresses. In this case, NAT is not a parameter of the network object; the network object or group is a parameter of the NAT configuration. The ability to use a network object *group* for the real address means that twice NAT is more scalable.
- How source and destination NAT is implemented.
  - Network object NAT— Each rule can apply to either the source or destination of a packet. So two rules might be used, one for the source IP address, and one for the destination IP address. These two rules cannot be tied together to enforce a specific translation for a source/destination combination.
  - Twice NAT—A single rule translates both the source and destination. A matching packet only matches the one rule, and further rules are not checked. Even if you do not configure the optional destination address for twice NAT, a matching packet still only matches one twice NAT rule. The source and destination are tied together, so you can enforce different translations depending on the source/destination combination. For example, sourceA/destinationA can have a different translation than sourceA/destinationB.
- Order of NAT Rules.
  - Network object NAT—Automatically ordered in the NAT table.
  - Twice NAT—Manually ordered in the NAT table (before or after network object NAT rules).

We recommend using network object NAT unless you need the extra features that twice NAT provides. Network object NAT is easier to configure, and might be more reliable for applications such as Voice over IP (VoIP). (For VoIP, because twice NAT is applicable only between two objects, you might see a failure in the translation of indirect addresses that do not belong to either of the objects.)

Information About Network Object NAT

All NAT rules that are configured as a parameter of a network object are considered to be *network object NAT* rules. Network object NAT is a quick and easy way to configure NAT for a network object, which can be a single IP address, a range of addresses, or a subnet.

After you configure the network object, you can then identify the mapped address for that object, either as an inline address or as another network object or network object group.

When a packet enters the ASA, both the source and destination IP addresses are checked against the network object NAT rules. The source and destination address in the packet can be translated by separate rules if separate matches are made. These rules are not tied to each other; different combinations of rules can be used depending on the traffic.

Because the rules are never paired, you cannot specify that sourceA/destinationA should have a different translation than sourceA/destinationB. Use twice NAT for that kind of functionality (twice NAT lets you identify the source and destination address in a single rule).

Information About Twice NAT

Twice NAT lets you identify both the source and destination address in a single rule. Specifying both the source and destination addresses lets you specify that sourceA/destinationA can have a different translation than sourceA/destinationB.

The destination address is optional. If you specify the destination address, you can either map it to itself (identity NAT), or you can map it to a different address. The destination mapping is always a static mapping.

Twice NAT also lets you use service objects for static NAT with port translation; network object NAT only accepts inline definition.

Configuring Identity NAT

Identity NAT translates the real IP address to the same IP address. Only "translated" hosts can create NAT translations, and responding traffic is allowed back.

Secure Copy

The Secure Copy (SCP) feature provides a secure and authenticated method for copying device configurations or device image files. SCP relies on Secure Shell (SSH), an application and protocol that provide a secure replacement for the Berkeley r-tools suite (Berkeley university's own set of networking applications). This document provides the procedure to configure a Cisco device for SCP server-side functionality.

Prerequisites for Secure Copy

- **Before enabling Secure Copy (SCP), you must correctly configure Secure Shell (SSH)**, authentication, and authorization on the device.
- Because SCP relies on SSH for its secure transport, the device must have a Rivest, Shamir, and Adelman (RSA) key pair.

Information About Secure Copy

How Secure Copy Works

The behavior of Secure Copy (SCP) is similar to that of remote copy (RCP), which comes from the Berkeley r-tools suite (Berkeley university's own set of networking applications), except that SCP relies on Secure Shell (SSH) for security. In addition, SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so that the device can determine whether the user has the correct privilege level.

SCP allows a user with appropriate authorization to copy any file that exists in the Cisco IOS FIle System (IFS) to and from a device by using the **copy** command. An authorized administrator may also perform this action from a workstation.

NTP Overview

NTP is designed to synchronize the time on a network of machines. NTP runs over the User Datagram Protocol (UDP), using port 123 as both the source and destination, which in turn runs over IP. NTP Version 3 RFC 1305 ↗ is used to synchronize timekeeping among a set of distributed time servers and clients. A set of nodes on a network are identified and configured with NTP and the nodes form a

synchronization subnet, sometimes referred to as an overlay network. While multiple masters (primary servers) may exist, there is no requirement for an election protocol.
An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. An NTP client makes a transaction with its server over its polling interval (from 64 to 1024 seconds) which dynamically changes over time depending on the network conditions between the NTP server and the client. The other situation occurs when the router communicates to a bad NTP server (for example, NTP server with large dispersion); the router also increases the poll interval. No more than one NTP transaction per minute is needed to synchronize two machines. It is not possible to adjust the NTP poll interval on a router.

NTP uses the concept of a stratum to describe how many NTP hops away a machine is from an authoritative time source. For example, a stratum 1 time server has a radio or atomic clock directly attached to it. It then sends its time to a stratum 2 time server through NTP, and so on. A machine running NTP automatically chooses the machine with the lowest stratum number that it is configured to communicate with using NTP as its time source. This strategy effectively builds a self-organizing tree of NTP speakers. NTP performs well over the non-deterministic path lengths of packet-switched networks, because it makes robust estimates of the following three key variables in the relationship between a client and a time server.

- Network delay
- Dispersion of time packet exchanges—A measure of maximum clock error between the two hosts.
- Clock offset—The correction applied to a client's clock to synchronize it.

Clock synchronization at the 10 millisecond level over long distance wide-area networks (WANs) (2000 km), and at the 1 millisecond level for local-area networks (LANs), is routinely achieved.

NTP avoids synchronizing to a machine whose time may not be accurate in two ways. First of all, NTP never synchronizes to a machine that is not synchronized itself. Secondly, NTP compares the time reported by several machines, and will not synchronize to a machine whose time is significantly different than the others, even if its stratum is lower.

The communications between machines running NTP (associations) are usually statically configured. Each machine is given the IP address of all machines with which it should form associations. Accurate timekeeping is made possible by exchanging NTP messages between each pair of machines with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each machine can be configured to send or receive broadcast messages. However, the accuracy of timekeeping is marginally reduced because the information flow is one-way only.

The time kept on a machine is a critical resource and it is strongly recommend that you use the security features of NTP to avoid the accidental or malicious setting of incorrect time. **The two security features available are an access list-based restriction scheme and an encrypted authentication mechanism.**

Cisco's implementation of NTP supports the stratum 1 service in certain Cisco IOS software releases. If a release supports the **ntp refclock** command, it is possible to connect a radio or atomic clock. Certain releases of Cisco IOS support either the Trimble Palisade NTP Synchronization Kit (Cisco 7200 series routers only) or the Telecom Solutions Global Positioning System (GPS) device. If the network uses the public time servers on the Internet and the network is isolated from the Internet, Cisco's implementation of NTP allows a machine to be configured so that it acts as though it is synchronized through NTP, when in fact it has determined the time using other means. Other machines then synchronize to that machine through NTP.

RFC 6353

Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)

Abstract

This document describes a **Transport Model** for the Simple Network Management Protocol (SNMP), that **uses either the Transport Layer Security protocol or the Datagram Transport Layer Security (DTLS) protocol. The TLS and DTLS protocols provide authentication and privacy services for SNMP applications.** This document describes how the TLS Transport Model (TLSTM) implements the needed features of an SNMP Transport Subsystem to make this protection possible in an interoperable way.

This Transport Model is designed to meet the security and operational needs of network administrators. It supports the sending of SNMP messages over TLS/TCP and DTLS/UDP. The TLS mode can make use of TCP's improved support for larger packet sizes and the DTLS mode provides potentially superior operation in environments where a connectionless (e.g., UDP) transport is preferred. Both TLS and DTLS integrate well into existing public keying infrastructures. This document also defines a portion of the Management Information Base (MIB) for use with network management protocols. In particular, it defines objects for managing the TLS Transport Model for SNMP.

Q and A from support forum on Cisco.com regarding Cisco AnyConnect

Q. Can the AnyConnect client work through an IPsec VPN remote access client tunnel (tunnel-over-tunnel) , or vice versa?
A. This is not officially supported. The reason it cannot work is because both the IPsec client and the AnyConnect client are trying to route traffic to their virtual adapters. The IPsec client is intercepting AnyConnect traffic at the IM layer.

Note:Clientless SSL VPN traffic can pass over a full-tunnel remote access client (AnyConnect or IPSec) and Site to Site IPSec.

Q. How does the AnyConnect client enforce/monitor the tunnel/split-tunnel policy?
A. AnyConnect enforces the tunnel policy in 2 ways:

1)Route monitoring and repair (e.g. if you change the route table), AnyConnect will restore it to what was provisioned.

2)Filtering (on platforms that support filter engines). Filtering ensures that even if you could perform some sort of route injection, the filters would block the packets.

Q. Can AnyConnect (or Clientless SSL VPN) users "initiate" password-management/changes from the AnyConnect client itself?
A. No. AnyConnect does not have any option inside of it to trigger or initate a password change.

Password changes are only triggered from the head-end when required as part of MS-CHAPv2 RADIUS with expiry or Lightweight Directory Access Protocol (LDAP) password expiration. Customers can change their Active Directory (AD) password using the same ctrl-alt-del mechanism assuming they are ‘logging in to the network’ (Start Before Login).

Q. Does AnyConnect support a pool with a single address? If you want the ASA to do Port Address Translation (PAT), such that all the remote clients appear on the inside network as a single address, differentiated by source TCP port number?
A. AnyConnect SSL VPN client , like a n IPSec full-tunnel client, requires a unique IP address for each client. Thus, the PAT pool does not apply with AnyConnect in this context. Certainly, going through a Linksys/IOS 871 router/ASA 5505 which does PAT is not an issue with AnyConnect.

Q. Does AnyConnect have the ability to be able to present a popup with the list of certificates, such as what is available for SSL VPN Clientless?
A. There is no popup asking the user for certificate selection. The enhancement for this capability is tracked via CSCsk56537. As an immediate solution, the administrator can specify certificate match selection criteria in the AnyConnect Profile XML file. Refer to Configuring the Certificate Match Attribute.

Configuring DTLS

Datagram Transport Layer Security (DTLS) allows the AnyConnect client establishing an SSL VPN connection to use two simultaneous tunnels—an SSL tunnel and a DTLS tunnel. Using DTLS avoids latency and bandwidth problems associated with SSL connections and improves the performance of real-time applications that are sensitive to packet delays.

In order for DTLS to fall back to a TLS connection, Dead Peer Detection (DPD) must be enabled. If you do not enable DPD, and the DTLS connection experiences a problem, the connection terminates instead of falling back to TLS.

Enabling and Adjusting Dead Peer Detection

Dead Peer Detection (DPD) ensures that the ASA (gateway) or the client can quickly detect a condition where the peer is not responding, and the connection has failed.

SSL/TLS

Secure Sockets Layer (SSL) is an **application-layer protocol** that provides encryption technology for the Internet. SSL ensures the secure transmission of data between a client and a server through a combination of privacy, authentication, and data integrity. SSL relies upon certificates and private-public key exchange pairs for this level of security.

SSL facilitates this authentication through the use of **digital certificates**. Digital certificates are a form of digital identification to prove the identity of the server to the client, or optionally, the client to the server. A certificate ensures that the identification information is correct and that the public key embedded in the certificate actually belongs to that client or server.

The client and server use the **SSL handshake protocol** to establish an SSL session between the two devices. During the handshake, the client and server negotiate the SSL parameters that they will use during the secure session.

Port Security

Port Security **mitigates against Layer 2 MAC address table overflow attacks** (also known as CAM overflow), which exhaust switch hardware CAM tables by bombarding the switch with random MAC addresses, so that new host MACs are flooded in the network, thus potentially slowing network performance and increasing CPU load on clients and hosts. Port Security can limit the number of MAC addresses learned on a particular port. Using this feature, hosts cannot overload the CAM tables with more than the configured amount of MAC addresses for their port.

Enabling TCP Intercept

To enable TCP intercept, use the following commands in global configuration mode:

	Command	Purpose
Step 1	`Router(config)# access-list access-list-number {deny | permit} tcp any destination destination-wildcard`	Defines an IP extended access list.
Step 2	`Router(config)# ip tcp intercept list access-list-number`	Enables TCP intercept.

You can define an access list to intercept all requests or only those coming from specific networks or destined for specific servers. Typically the access list will define the source as **any** and define specific destination networks or servers. That is, you do not attempt to filter on the source addresses because you do not necessarily know who to intercept packets from. You identify the destination in order to protect destination servers.

If no access list match is found, the router allows the request to pass with no further action.

Setting the TCP Intercept Mode

The TCP intercept can operate in either active intercept mode or passive watch mode. The default is intercept mode.

In intercept mode, the software actively intercepts each incoming connection request (SYN) and responds on behalf of the server with an SYN-ACK, then waits for an ACK from the client. When that ACK is received, the original SYN is sent to the server and the software performs a three-way handshake with the server. When this is complete, the two half-connections are joined.

In watch mode, connection requests are allowed to pass through the router to the server but are watched until they become established. If they fail to become established within 30 seconds (configurable with the **ip tcp intercept watch-timeout** command), the software sends a Reset to the server to clear up its state.

To set the TCP intercept mode, use the following command in global configuration mode:

Command	Purpose
`Router(config)# ip tcp intercept mode {intercept | watch}`	Sets the TCP intercept mode.

New Cisco ASA user account

When creating a new user account, the user name keyword is a string from 4 to 64 characters long.

The password keyword is a string from 3 to 32 characters long. The privilege level argument sets the privilege level, which ranges from 0 to 15. **The default is 2**. This privilege level is used with command authorization.

Caution If you do not use command authorization (the "**aaa authorization console LOCAL**" command), then the default level 2 allows management access to privileged EXEC mode. If you want to limit access to privileged EXEC mode, either set the privilege level to 0 or 1, or use the service-type command.

ASA and TACACS+

The ASA supports TACACS+ server authentication with the following protocols: ASCII, PAP, CHAP, and MS-CHAPv1. Using TACACS+ Attributes. The ASA provides support for TACACS+ attributes. TACACS+ attributes separate the functions of authentication, authorization, and accounting. The protocol supports two types of attributes: mandatory and optional. Both the server and client must understand a mandatory attribute, and the mandatory attribute must be applied to the user. An optional attribute may or may not be understood or used.

ASA Syslog

You can configure the ASA and ASASM to send data to a syslog server using either UDP or TCP, but not both. The default protocol is UDP if you do not specify a protocol.

You can use multiple **logging host** commands to specify additional servers that would all receive syslog messages. If you configure two or more logging servers, make sure that you limit the logging severity level to warnings for all logging servers.

If you specify TCP, the ASA discovers when the syslog server fails and as a security protection, new connections through the ASA are blocked. If you specify UDP, the ASA continues to allow new connections whether or not the syslog server is operational. Valid port values for either protocol are 1025 through 65535. **The default UDP port is 514. The default TCP port is 1470**.

Including the Date and Time in Syslog Messages

To include the date and time in syslog messages, enter the following command:

Command	Purpose
`logging timestamp`	Specifies that syslog messages should include the date and time that they were generated. To remove the date and time from syslog messages, enter the **no logging timestamp** command.

`ciscoasa(config)# logging timestamp`
`ciscoasa(config)# logging timestamp LOG-2015-10-24-081856.TXT`	

aaa accounting

To enable authentication, authorization, and accounting (AAA) accounting of requested services for billing or security purposes when you use RADIUS or TACACS+, use the **aaa accounting** command in global configuration mode. To disable AAA accounting, use the **no** form of this command.

aaa accounting {**auth-proxy** | **system | network** | **exec** | **connection** | **commands** ***level***} {**default** | ***list-name***} {**start-stop** | **stop-only** | **none**}[**broadcast**] **group** *groupname*

no aaa accounting {**auth-proxy** | **system** | **network** | **exec** | **connection** | **commands** ***level***} {**default** | ***list-name***} [**broadcast**] **group groupname**

Syntax Description

auth-proxy	Provides information about all authenticated-proxy user events.
system	Performs accounting for all system-level events not associated with users, such as reloads.
network	Runs accounting for all network-related service requests, including SLIP[1] , PPP[2] , PPP NCPs[3] , and ARAP[4].
exec	Runs accounting for EXEC shell session. This keyword might return user profile information such as what is generated by the **autocommand** command.
connection	Provides information about all outbound connections made from the network access server, such as Telnet, LAT[5] , TN3270, PAD[6] , and rlogin.
commands*level*	Runs accounting for all commands at the specified privilege level. Valid privilege level entries are integers from 0 through 15.
default	Uses the listed accounting methods that follow this argument as the default list of methods for accounting services.
list-name	Character string used to name the list of at least one of the accounting methods.
start-stop	Sends a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins

	regardless of whether the "start" accounting notice was received by the accounting server.
stop-only	Sends a "stop" accounting notice at the end of the requested user process.
none	Disables accounting services on this line or interface.
broadcast	(Optional) Enables sending accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, fail over occurs using the backup servers defined within that group.
group*groupname*	At least one of the keywords.

Security Manager

FlexConfig policies allow you to configure device commands that are not otherwise supported by Security Manager. By using Flexconfigs, you can extend Security Manager's control over a device configuration and take advantage of new device features before upgrading the product.

FlexConfig policies are made up of FlexConfig objects. These objects are essentially subroutines that can include scripting language commands, device commands, and variables. You can configure an object to be processed prior to applying the Security Manager configuration to a device, or you can have it processed after the configuration. Security Manager processes your objects in the order you specify so that you can create objects whose processing depends on the processing of another object. A FlexConfig policy object's contents can range from a single simple command string to elaborate CLI command structures that incorporate scripting and variables.

Settings-Based Policies vs. Rule-Based Policies

Security Manager policies are structured as either rule-based policies or settings-based policies**. Rule-Based Policies Rule-based policies contain one or more rules that govern how to handle traffic on a selected device, such as the access rules and inspection rules defined as part of a firewall service.** Rule-based policies can contain hundreds or even thousands of rules arranged in a table, each defining different values for the same set of parameters. The ordering of the rules is very important, as traffic flows are assigned the first rule whose definition matches the flow (known as first matching). The structure of the rules table depends on whether you configure a local policy or a shared policy (see Local Policies vs. Shared Policies, page 6-3). If you configure a local rule-based policy for a single device, the policy contains a flat table of local rules. If you configure a shared rule-based policy (either in Device view or Policy view), the table is divided into two sections, Mandatory and Default. Mandatory rules always precede the default rules. You can define rules in either section and move rules between sections using cut-and-paste. When you define certain types of rule-based policies, such as firewall service policies, you can create a policy hierarchy in which rules located at lower levels in the hierarchy acquire properties from the rules located above them. This is known as rule inheritance. For example, you can define a set of inspection rules that apply globally to all firewalls, while supplementing these rules with additional rules that can be applied to a subset of devices. By maintaining common rules in a parent policy, inheritance enables you to reduce the chance of introducing configuration errors that will cause deployment to fail. For more information, see Understanding Rule Inheritance, page 6-4. Settings-Based

Policies Settings-based policies contain sets of related parameters that together define one aspect of security or device operation. For example, when you configure a Cisco IOS router, you can define a quality of service (QoS) policy that defines which interfaces are included in the policy, the type of traffic on which QoS is applied, and the definition of how this traffic should be queued and shaped. Unlike rule-based policies, which can contain hundreds of rules containing values for the same set of parameters, you can define only one set of parameters for each settings-based policy defined on a device.

IOS Role-Based CLI Access

The Role-Based CLI Access feature allows the network administrator to define "views," which are a set of operational commands and configuration capabilities that provide selective or partial access to Cisco IOS EXEC and configuration (Config) mode commands. Views restrict user access to Cisco IOS command-line interface (CLI) and configuration information; that is, a view can define what commands are accepted and what configuration information is visible. Thus, network administrators can exercise better control over access to Cisco networking devices.

Restrictions for Role-Based CLI Access

Lawful Intercept Images Limitation

Because CLI views are a part of the Cisco IOS parser, CLI views are a part of all platforms and Cisco IOS images. However, the lawful intercept view is available only in images that contain the lawful intercept subsystem.

Maximum Number of Allowed Views

The maximum number of CLI views and superviews, including one lawful intercept view, that can be configured is 15. (This does not include the root view.)

CCP for managing IOS

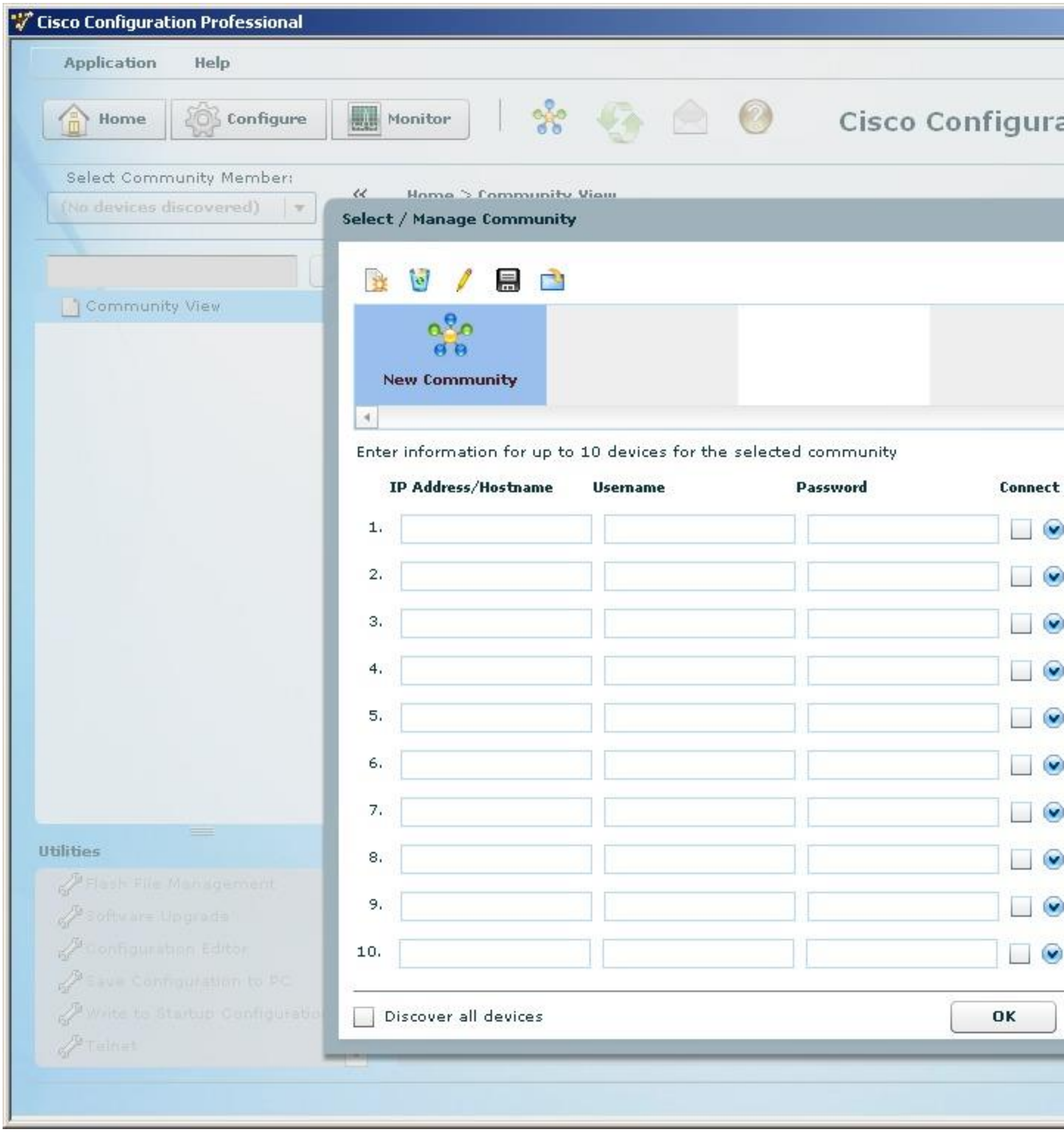
Be prepared to know how to navigate and use Cisco Configuration Professional (CCP) to discover, verify and/or configure elements of a Cisco router configuration, including zone based firewalls, ACLs, NTP, NAT, viewing IP addresses and interfaces, etc. Please refer to the “Cisco Configuration Professional User Guide Version 2.7” for additional details regarding using CCP.

Creating a Community and Adding Devices

Procedure

Use this procedure to create a community, add devices to it, and discover all the devices in a community.

Step 1 Use the Manage Community dialog box to create communities. The Manage Community dialog box automatically displays when you start Cisco CP and a community called, New Community, is created by default.

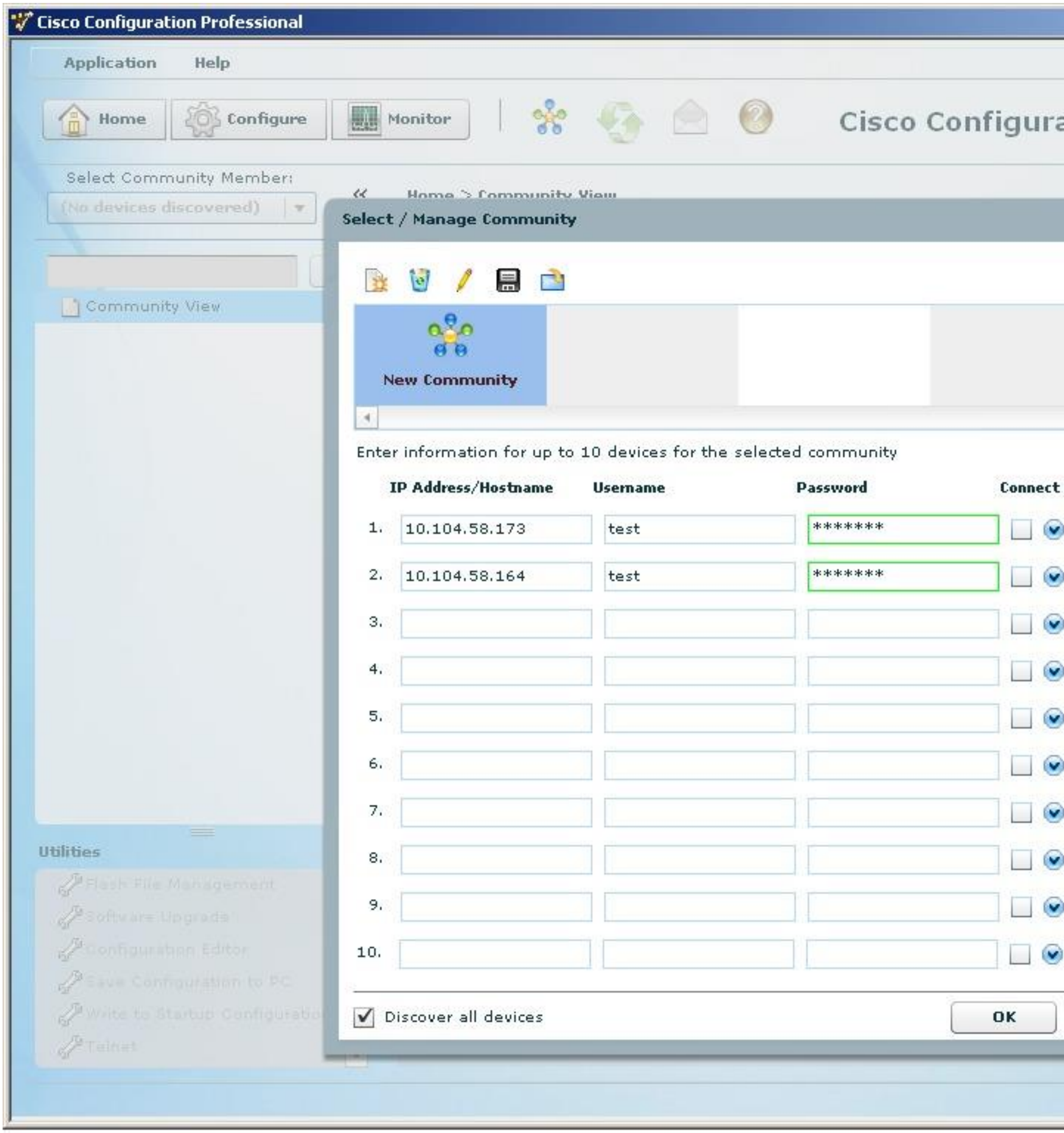


Enter information for up to 10 devices for the selected community

You can also open the Manage Community dialog box in the following ways:

- From the toolbar, click the **Manage Community** icon.
- From the menu bar, choose **Application > Manage Community**.

Step 2 In the Manage Community dialog box, enter the IP address or hostname; and the username and password information for the devices that you want to configure.



Enter information for up to 10 devices for the selected community

	IP Address/Hostname	Username	Password	Connect
1.	10.104.58.173	test	*******	☐
2.	10.104.58.164	test	*******	☐
3.				☐
4.				☐
5.				☐
6.				☐
7.				☐
8.				☐
9.				☐
10.				☐

☑ Discover all devices

If you enter the default username **cisco** and default password **cisco**, the Change Default Credentials dialog box opens. For security reasons, you must change the default credentials to new credentials.

Step 3 If you want Cisco CP to connect securely with the device, check the **Connect Securely** check box.

When you check the Connect Securely check box, HTTPS port 443 and SSH port 22 information is automatically added to the device. To view the port information, click the down-arrow next to the Connect Securely check box.

If you did not check the Connect Securely check box, the HTTP port 80 and Telnet port 23 information is automatically added to the device. To view the port information, click the down-arrow next to the Connect Securely check box.

Step 4 If you want to change the default port information, click it, and then enter a new port value.

Note Make sure that Cisco CP can access the device at the specified secure or non-secure ports.

Step 5 If you want Cisco CP to discover all the devices in a community, check the **Discover All Devices** check box. If you want, you can choose to discover the devices later, from the Community View page.

Step 6 Click **OK**. The Community View page opens. It displays the information about the devices in the community.

Understanding SPAN on a Catalyst Switch

You can analyze network traffic passing through ports by using SPAN to send a copy of the traffic to another port on the switch that has been connected to a SwitchProbe device or other Remote Monitoring (RMON) probe or security device. SPAN mirrors received or sent (or both) traffic on one or more source ports to a destination port for analysis.

Only traffic that enters or leaves source ports can be monitored by using SPAN.

SPAN does not affect the switching of network traffic on source ports; a copy of the packets received or sent by the source interfaces is sent to the destination interface. Except for traffic that is required for the SPAN session, reflector ports and destination ports do not receive or forward traffic.

You can use the SPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) Sensor Appliance to a destination port, the IDS device can send TCP Reset packets to close down the TCP session of a suspected attacker.

SPAN Session

A local SPAN session is an association of a local destination port with local source ports. You can monitor incoming or outgoing traffic on a series or range of ports.

SPAN sessions do not interfere with the normal operation of the switch. However, an oversubscribed SPAN destination, for example, a 10-Mbps port monitoring a 100-Mbps port, results in dropped or lost packets.

You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port for that session. The **show monitor session** *session_number* privileged EXEC command displays the operational status of a SPAN session.

A SPAN session remains inactive after system power-on until the destination port is operational.

show crypto session

To display status information for active crypto sessions, use the **show crypto session** command in privileged EXEC mode.

show crypto session [**groups** | **interface** *type* [**brief | detail**] | **isakmp** [**group** *group-name* | **profile** *profile-name*] [**brief | detail**] | [**local | remote**] [*ip-address* | *ipv6-address*] [**port** *port-number*] | [**fvrf** *fvrf-name*] [**ivrf** *ivrf-name*] [**brief | detail**] | **summary** *group-name* | **username** *username*]

IPsec and IKE Stateful Failover Syntax

show crypto session [**active | standby**]

Syntax Description

groups	(Optional) Displays crypto session group usage for all groups.
interface *type*	(Optional) Displays crypto sessions on the connected interface. - The *type* value is the type of interface connection.
brief	(Optional) Provides brief information about the session, such as the peer IP address, interface, username, group name or phase1 ID, length of session uptime, and current session status (up/down).
detail	(Optional) Provides detailed information about the session, such as the capability of the Internet Key Exchange (IKE) security association (SA), connection ID, remaining lifetime of the IKE SA, inbound or outbound encrypted or decrypted packet number of the IP security (IPsec) flow, dropped packet number, and kilobyte-per-second lifetime of the IPsec SA.

isakmp group *group-name*	(Optional) Displays crypto sessions using the Internet Security Association and Key Management Protocol (ISAKMP) group. • The *group-name* value is the name of the group.
isakmp profile *profile-name*	(Optional) Displays crypto sessions using the ISAKMP profile. • The *profile-name* value is the name of the profile.
local	(Optional) Displays status information about crypto sessions of a local crypto endpoint.
remote	(Optional) Displays status information about crypto sessions of a remote session.
ip-address	IP address of the local or remote crypto endpoint.
ipv6-address	IPv6 address of the local or remote crypto endpoint.
port *port-number*	(Optional) Displays status information about the port of the local crypto endpoint. • The *port-number* value can be from 1 to 65535. The default value is 500.
fvrf *fvrf-name*	(Optional) Displays status information about the front door virtual routing and forwarding (fVRF) session. • The *fvrf-name* value is the name of the fVRF session.
ivrf *ivrf-name*	(Optional) Displays status information about the inside VRF (iVRF) session.

	- The *ivrf-name* value is the name of the iVRF session. **Note** The iVRF session can have an empty value when VRF-aware IPsec (fVRF and iVRF) uses IPsec protected tunnels sharing the same tunnel source and the same IPsec profile. This scenario is valid for the following conditions: - IPsec protected multipoint generic routing encapsulation (mGRE) - IPsec protected Point-to-Point GRE tunnels
summary *group-name*	(Optional) Displays a list of crypto session groups and associated group members.
username *username*	(Optional) Displays the crypto session for the specified extended authentication (XAUTH), public key infrastructure (PKI), or authentication, authorization, and accounting (AAA) username.
active	(Optional) Displays all crypto sessions in the active state.
standby	(Optional) Displays all crypto sessions that are in the standby state.

Command Default

When no optional keywords and arguments are specified, all existing sessions are displayed.

IPsec Transport Mode and Tunnel Mode

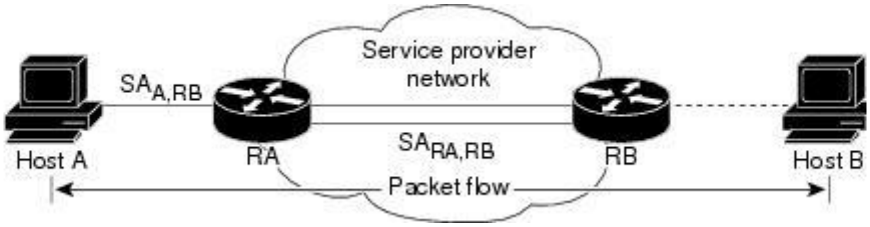
IPsec supports two encryption modes: *Transport mode* and *Tunnel mode*. *Transport mode* encrypts only the data portion (payload) of each packet and leaves the packet header untouched. Transport mode is applicable to either gateway or host implementations, and provides protection for upper layer protocols as well as selected IP header fields.

Tunnel mode **is more secure than Transport mode because it encrypts both the payload and the header**. IPsec in Tunnel mode is normally used when the ultimate destination of a packet is different than the security termination point. This mode is also used in cases when the security is provided by a device that did not originate packets, as in the case of VPNs.

Tunnel mode is often used in networks with unregistered IP addresses. The unregistered address can be tunneled from one gateway encryption device to another by hiding the unregistered addresses in the tunneled packet.

Figure 1-3 shows a typical network using IPsec in Tunnel mode:

Figure 1-3 IPsec in Tunnel Mode



In Tunnel mode, IPsec encapsulates an IP packet with IPsec headers and adds an outer IP header

An IPsec Tunnel mode packet has two IP headers—an inner header and an outer header. The inner header is constructed by the host; the outer header is added by the device that is providing security services. IPsec defines Tunnel mode for both the Authentication Header (AH) and Encapsulating Security Payload (ESP).

IPsec standards define several new packet formats, such as an Authentication Header (AH) to provide data integrity and the Encapsulating Security Payload (ESP) to provide confidentiality. IPsec parameters between devices are negotiated with the Internet Key Exchange (IKE) protocol, formerly referred to as the Internet Security Association Key Management Protocol (ISAKMP/Oakley).

IKE can use digital certificates for device authentication. The Encapsulating Security Payload and the Authentication Header use cryptographic techniques to ensure data confidentiality and digital signatures that authenticate the data's source.

The IP packet is the fundamental unit of communications in IP networks. IPsec handles encryption at the packet level, and the protocol it uses is the ESP. ESP supports any type of symmetric encryption. The default standard built into ESP that assures basic interoperability is 56-bit DES.

Using IPsec to Secure the IP Layer

IPsec is a framework of open standards developed by the Internet Engineering Task Force (IETF) that provides security for transmission of sensitive information over unprotected networks such as the Internet. It acts at the network level and implements the following standards:

- IPsec
- Internet Key Exchange (IKE)
- Data Encryption Standard (DES)
- MD5 (HMAC variant)
- SHA (HMAC variant)
- Authentication Header (AH)
- Encapsulating Security Payload (ESP)

Essentially, if the IPsec suite is used where IP is normally used (in the network layer), communications are secured for all applications and for all users more transparently than would be the case if any other approach was employed. With IPsec, a service provider can create a secure VPN as needed and with any other device that is using the IPsec standard. Because IPsec works with both existing and future IP standards, regular IP networks can still be used to carry data. The sending and receiving devices must be IPsec compliant, but the rest of the network between the sender and recipient does not have to be IPsec compliant.

The primary strength of the IPsec approach is that security works at a low network level. As a result, IP is transparent to the average user, and IPsec-based security services also function behind the scenes to ensure that all network communications are secure. IPsec meets a broad range of security needs and allows different networks around the world to interconnect and to communicate securely. In addition, IPsec offers almost infinite scalability with transparent and reliable service, no matter how demanding a company's security needs.

Cisco IPsec Technologies

Cisco IPsec includes the following technologies:

- IPsec

IPsec uses encryption technology to provide data confidentiality, integrity, and authenticity between participating peers in a private network. Cisco provides full Encapsulating Security Payload (ESP) and Authentication Header (AH) support.

- Internet Key Exchange (IKE)

The Internet Key Exchange (IKE) provides security association management. IKE authenticates each peer in an IPsec transaction, negotiates security policy, and handles the exchange of session keys. Cisco has been leading the standardization effort for IKE by writing IETF Internet drafts and by making a freeware version of IKE available on the Internet.

- Certificate management

Cisco supports the X509.V3 certificates for device authentication during IKE negotiation. Certificate management includes the use of the Simple Certificate Enrollment Protocol (SCEP), a protocol for communicating with Certification Authorities (CA). This certificate solution supports hierarchical certificate structures and the cross-certification necessary for a public key infrastructure (PKI) solution.

The component technologies include the following:

- Diffie-Hellman

Diffie-Hellman is a public-key cryptography protocol that allows two parties to establish a shared secret over an unsecure communications channel. IKE uses Diffie-Hellman to establish session keys. VPN Solutions Center supports two Diffie-Hellman groups: Group 1—a MODP group with a 768-bit modulus; Group 2—a MODP group with a 1024-bit modulus.

- AES

The Advanced Encryption Standard (AES) encrypts packet data.

- MD5/SHA algorithms

The Message Digest 5/SHA hash algorithms authenticate packet data.

STP Overview

STP is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Switches might also learn end-station MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one switch of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The switch that has *all* of its ports as the designated role or as the backup role is the root switch. The switch that has at least *one* of its ports in the designated role is called the designated switch.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Switches send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The switches do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending switch and its ports, including switch and MAC addresses, switch priority, port priority, and path cost. Spanning tree uses this information to elect the root switch and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a switch are part of a loop, the spanning-tree port priority and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (switch priority and MAC address) associated with each VLAN on each switch.
- The spanning-tree path cost to the root switch.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the switches in a network are powered up, each functions as the root switch. Each switch sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the switch that the sending switch identifies as the root switch
- The spanning-tree path cost to the root
- The bridge ID of the sending switch
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a switch receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the switch, the switch also forwards it with an updated message to all attached LANs for which it is the designated switch.

If a switch receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the switch is a designated switch for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- **One switch in the network is elected as the root switch (the logical center of the spanning-tree topology in a switched network).**

  For each VLAN, the sw itch with the highest switch priority (the lowest numerical priority value) is elected as the root switch. **If all switches are configured with the default priority (32768), the switch with the lowest MAC address in the VLAN becomes the root switch.** The switch priority value occupies the most significant bits of the bridge ID.
- A root port is selected for each switch (except the root switch). This port provides the best path (lowest cost) when the switch forwards packets to the root switch.
- The shortest distance to the root switch is calculated for each switch based on the path cost.
- A designated switch for each LAN segment is selected. The designated switch incurs the lowest path cost when forwarding packets from that LAN to the root switch. The port through which the designated switch is attached to the LAN is called the designated port.

All paths that are not needed to reach the root switch from anywhere in the switched network are placed in the spanning-tree blocking mode.

VTP Modes

You can configure a switch to operate in any one of these VTP modes:

- Server—In VTP server mode, you can create, modify, and delete VLANs and specify other configuration parameters, such as VTP version and VTP pruning, for the entire VTP domain. VTP servers advertise their VLAN configuration to other switches in the same VTP domain and synchronize their VLAN configuration with other switches based on advertisements received over trunk links. VTP server is the default mode.
- Client—VTP clients behave the same way as VTP servers, but you cannot create, change, or delete VLANs on a VTP client.
- Transparent—VTP transparent switches do not participate in VTP. A VTP transparent switch does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements, but transparent switches do forward VTP advertisements that they receive out their trunk ports in VTP Version 2.

Impact of mismatching Native VLANs on a trunk:

https://supportforums.cisco.com/discussion/10423901/impact-native-vlna-mismatch

Understanding DAI and ARP Spoofing Attacks

Dynamic ARP Inspection (DAI) is a security feature that validates ARP packets in a network. DAI intercepts, logs, and **discards ARP packets with invalid IP-to-MAC address bindings**. This capability protects the network from some man-in-the-middle attacks.

DAI ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

DAI determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

DAI can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses.

You can configure DAI to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header.

Guidelines on placement of ACLs

https://supportforums.cisco.com/discussion/9426506/conflicting-advice-acl-placement

Information About Implementing Management Plane Protection

Inband Management Interface

An *inband management interface* is a Cisco IOS XR software physical or logical interface that processes management packets, as well as data-forwarding packets. An inband management interface is also called a *shared management interface*.

Out-of-Band Management Interface

Out-of-band refers to an interface that allows only management protocol traffic to be forwarded or processed. An *out-of-band management interface* is defined by the network operator to specifically receive network management traffic. The advantage is that forwarding (or customer) traffic cannot interfere with the management of the router, which significantly reduces the possibility of denial-of-service attacks.

Out-of-band interfaces forward traffic only between out-of-band interfaces or terminate management packets that are destined to the router. In addition, the out-of-band interfaces can participate in dynamic routing protocols. The service provider connects to the router's out-of-band interfaces and builds an independent overlay management network, with all the routing and policy tools that the router can provide.

Peer-Filtering on Interfaces

The peer-filtering option allows management traffic from specific peers, or a range of peers, to be configured.

Control Plane Protection Overview

A *control plane* is a collection of processes that run at the process level on a route processor and collectively provide high-level control for most Cisco IOS XR software functions. All traffic directly or indirectly destined to a router is handled by the control plane. Management Plane Protection operates within the Control Plane Infrastructure.

Management Plane

The *management plane* is the logical path of all traffic that is related to the management of a routing platform. One of three planes in a communication architecture that is structured in layers and planes, the management plane performs management functions for a network and coordinates functions among all the planes (management, control, and data). In addition, the management plane is used to manage a device through its connection to the network.

Examples of protocols processed in the management plane are Simple Network Management Protocol (SNMP), Telnet, HTTP, Secure HTTP (HTTPS), and SSH. These management protocols are used for monitoring and for command-line interface (CLI) access. Restricting access to devices to internal sources (trusted networks) is critical.

Management Plane Protection Feature

The MPP protection feature, as well as all the management protocols under MPP, are disabled by default. When you configure an interface as either out-of-band or inband, it automatically enables MPP. Consequently, this enablement extends to all the protocols under MPP.

If MPP is disabled and a protocol is activated, all interfaces can pass traffic.

When MPP is enabled with an activated protocol, the only default management interfaces allowing management traffic are the route processor (RP) and standby route processor (SRP) Ethernet interfaces. You must manually configure any other interface for which you want to enable MPP as a management interface, using the MPP CLI that follows. Afterwards, only the default management interfaces and those you have previously configured as MPP interfaces will accept network management packets destined for the device. All other interfaces drop such packets.

After configuration, you can modify or delete a management interface.

Following are the management protocols that the MPP feature supports. These management protocols are also the only protocols affected when MPP is enabled.

- SSH, v1 and v2
- SNMP, all versions
- Telnet
- TFTP
- HTTP
- HTTPS

Benefits of the Management Plane Protection Feature

- Greater access control for managing a device than allowing management protocols on all interfaces.
- Improved performance for data packets on non-management interfaces.
- Support for network scalability.
- Simplifies the task of using per-interface access control lists (ACLs) to restrict management access to the device.
- Fewer ACLs are needed to restrict access to the device.
- Prevention of packet floods on switching and routing interfaces from reaching the CPU.