



Security Configuration Guide, Cisco IOS XE Everest 16.5.1a (Catalyst 3850 Switches)

First Published: 2017-05-31

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



CONTENTS

CHAPTER 1

Preventing Unauthorized Access	1
Finding Feature Information	1
Restrictions for Controlling Switch Access with Passwords and Privileges	1
Information About Passwords and Privilege Levels	2
Default Password and Privilege Level Configuration	2
Additional Password Security	2
Password Recovery	2
Terminal Line Telnet Configuration	3
Username and Password Pairs	3
Privilege Levels	3
How to Control Switch Access with Passwords and Privilege Levels	4
Setting or Changing a Static Enable Password	4
Protecting Enable and Enable Secret Passwords with Encryption	5
Disabling Password Recovery	7
Setting a Telnet Password for a Terminal Line	8
Configuring Username and Password Pairs	10
Setting the Privilege Level for a Command	11
Changing the Default Privilege Level for Lines	13
Logging into and Exiting a Privilege Level	14
Monitoring Switch Access	14
Configuration Examples for Setting Passwords and Privilege Levels	15
Example: Setting or Changing a Static Enable Password	15
Example: Protecting Enable and Enable Secret Passwords with Encryption	15
Example: Setting a Telnet Password for a Terminal Line	15
Example: Setting the Privilege Level for a Command	15
Additional References	16

CHAPTER 2	Controlling Switch Access with Passwords and Privilege Levels	17
	Finding Feature Information	17
	Restrictions for Controlling Switch Access with Passwords and Privileges	17
	Information About Passwords and Privilege Levels	18
	Default Password and Privilege Level Configuration	18
	Additional Password Security	18
	Password Recovery	18
	Terminal Line Telnet Configuration	19
	Username and Password Pairs	19
	Privilege Levels	19
	How to Control Switch Access with Passwords and Privilege Levels	20
	Setting or Changing a Static Enable Password	20
	Protecting Enable and Enable Secret Passwords with Encryption	21
	Disabling Password Recovery	23
	Setting a Telnet Password for a Terminal Line	24
	Configuring Username and Password Pairs	26
	Setting the Privilege Level for a Command	27
	Changing the Default Privilege Level for Lines	29
	Logging into and Exiting a Privilege Level	30
	Monitoring Switch Access	30
	Configuration Examples for Setting Passwords and Privilege Levels	31
	Example: Setting or Changing a Static Enable Password	31
	Example: Protecting Enable and Enable Secret Passwords with Encryption	31
	Example: Setting a Telnet Password for a Terminal Line	31
	Example: Setting the Privilege Level for a Command	31
	Additional References	32
CHAPTER 3	Configuring TACACS+	33
	Finding Feature Information	33
	Prerequisites for TACACS+	33
	Information About Controlling Switch Access with TACACS+	34
	TACACS+ and Switch Access	34
	TACACS+ Overview	34

TACACS+ Operation	36
Method List	36
TACACS+ Configuration Options	37
TACACS+ Login Authentication	37
TACACS+ Authorization for Privileged EXEC Access and Network Services	37
TACACS+ Accounting	37
Default TACACS+ Configuration	38
How to Configure Switch Access with TACACS+	38
Identifying the TACACS+ Server Host and Setting the Authentication Key	38
Configuring TACACS+ Login Authentication	40
Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services	42
Starting TACACS+ Accounting	43
Establishing a Session with a Router if the AAA Server is Unreachable	45
Monitoring TACACS+	45
Additional References For Switch Access with TACACS+	45
Feature Information for Switch Access with TACACS+	46

CHAPTER 4
Configuring RADIUS 47

Finding Feature Information	47
Prerequisites for Configuring RADIUS	47
Restrictions for Configuring RADIUS	48
Information about RADIUS	49
RADIUS and Switch Access	49
RADIUS Overview	49
RADIUS Operation	50
RADIUS Change of Authorization	51
Change-of-Authorization Requests	52
CoA Request Response Code	54
CoA Request Commands	55
Stacking Guidelines for Session Termination	57
Default RADIUS Configuration	58
RADIUS Server Host	58
RADIUS Login Authentication	59
AAA Server Groups	60

AAA Authorization	60
RADIUS Accounting	60
Vendor-Specific RADIUS Attributes	60
Vendor-Proprietary RADIUS Server Communication	72
How to Configure RADIUS	72
Identifying the RADIUS Server Host	72
Configuring RADIUS Login Authentication	75
Defining AAA Server Groups	77
Configuring RADIUS Authorization for User Privileged Access and Network Services	79
Starting RADIUS Accounting	80
Configuring Settings for All RADIUS Servers	81
Configuring the Device to Use Vendor-Specific RADIUS Attributes	83
Configuring the Device for Vendor-Proprietary RADIUS Server Communication	84
Configuring CoA on the Device	86
Monitoring CoA Functionality	88
Additional References for Configuring Secure Shell	88

CHAPTER 5**Configuring Kerberos 91**

Finding Feature Information	91
Prerequisites for Controlling Switch Access with Kerberos	91
Information about Kerberos	92
Kerberos and Switch Access	92
Kerberos Overview	92
Kerberos Operation	94
Authenticating to a Boundary Switch	94
Obtaining a TGT from a KDC	95
Authenticating to Network Services	95
How to Configure Kerberos	95
Monitoring the Kerberos Configuration	95
Additional References	95

CHAPTER 6**MACsec Encryption 97**

Finding Feature Information	97
Prerequisites for MACsec Encryption	97

Information About MACsec Encryption	98
Recommendations for MACsec Encryption	98
Media Access Control Security and MACsec Key Agreement	99
MKA Policies	99
Virtual Ports	99
MKA Statistics	100
Key Lifetime and Hitless Key Rollover	100
MACsec, MKA and 802.1x Host Modes	100
Understanding MACsec MKA using EAP-TLS	101
Prerequisites for MACsec MKA using EAP-TLS	101
Limitations for MACsec MKA using EAP-TLS	102
Cisco TrustSec Overview	102
How to Configure MACsec Encryption	103
Configuring MKA and MACsec	103
Default MACsec MKA Configuration	103
Configuring an MKA Policy	104
Configuring Switch-to-host MACsec Encryption	105
Configuring MACsec MKA using PSK	107
Configuring MACsec MKA on an Interface using PSK	107
Configuring MACsec MKA using EAP-TLS	109
Remote Authentication	109
Local Authentication	115
Verifying MACsec MKA using EAP-TLS	121
Configuring Cisco TrustSec MACsec	123
Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode	123
Configuration Examples for MACsec Encryption	125
Configuring Switch-to-host MACsec Encryption	125
Configuration Examples for MACsec MKA using EAP-TLS	127
Example: Enrolling the Certificate	127
Example: Enabling 802.1x Authentication and AAA Configuration	128
Example: Configuring EAP-TLS Profile and 802.1X Credentials	128
Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface	128
Example: Displaying MKA Information	128
Cisco TrustSec Switch-to-Switch Link Security Configuration Example	134

CHAPTER 7	Configuring Local Authentication and Authorization	137
	Finding Feature Information	137
	How to Configure Local Authentication and Authorization	137
	Configuring the Switch for Local Authentication and Authorization	137
	Monitoring Local Authentication and Authorization	139
	Additional References	140

CHAPTER 8	Configuring Secure Shell (SSH)	141
	Finding Feature Information	141
	Prerequisites for Configuring Secure Shell	141
	Restrictions for Configuring Secure Shell	142
	Information About Configuring Secure Shell	142
	SSH And Switch Access	142
	SSH Servers, Integrated Clients, and Supported Versions	143
	SSH Configuration Guidelines	143
	Secure Copy Protocol Overview	143
	Secure Copy Protocol	144
	How to Configure Secure Shell	144
	Setting Up the Device to Run SSH	144
	Configuring the SSH Server	146
	Monitoring the SSH Configuration and Status	148

CHAPTER 9	X.509v3 Certificates for SSH Authentication	149
	X.509v3 Certificates for SSH Authentication	149
	Prerequisites for Digital Certificates for SSH Authentication	149
	Restrictions for X.509v3 Certificates for SSH Authentication	149
	Information About X.509v3 Certificates for SSH Authentication	150
	Digital Certificates	150
	Server and User Authentication using X.509v3	150
	How to Configure X.509v3 Certificates for SSH Authentication	150
	Configuring IOS SSH Server to Use Digital Certificates for Sever Authentication	150
	Configuring IOS SSH Server to Verify User's Digital Certificate for User Authentication	152
	Verifying Configuration for Server and User Authentication Using Digital Certificates	154

Configuration Examples for X.509v3 Certificates for SSH Authentication	154
Example: Configuring IOS SSH Server to Use Digital Certificates for Server Authentication	154
Example: Configuring IOS SSH Server to Verify User's Digital Certificate for User Authentication	155
Additional References for X.509v3 Certificates for SSH Authentication	155
Feature Information for X.509v3 Certificates for SSH Authentication	156

CHAPTER 10

Configuring Secure Socket Layer HTTP	157
Finding Feature Information	157
Information about Secure Socket Layer HTTP	157
Secure HTTP Servers and Clients Overview	157
Certificate Authority Trustpoints	158
CipherSuites	159
Default SSL Configuration	160
SSL Configuration Guidelines	160
How to Configure Secure Socket Layer HTTP	161
Configuring a CA Trustpoint	161
Configuring the Secure HTTP Server	163
Configuring the Secure HTTP Client	166
Monitoring Secure HTTP Server and Client Status	167
Additional References for Secure Socket Layer HTTP	168
Feature Information for Secure Socket Layer HTTP	168

CHAPTER 11

Configuring IPv4 ACLs	171
Finding Feature Information	171
Restrictions for Configuring IPv4 Access Control Lists	171
Information about Network Security with ACLs	172
ACL Overview	173
Access Control Entries	173
ACL Supported Types	173
Supported ACLs	173
ACL Precedence	174
Port ACLs	174
Router ACLs	175

VLAN Maps	176
ACEs and Fragmented and Unfragmented Traffic	176
ACEs and Fragmented and Unfragmented Traffic Examples	176
ACLs and Switch Stacks	177
Active Switch and ACL Functions	177
Stack Member and ACL Functions	178
Active Switch Failure and ACLs	178
Standard and Extended IPv4 ACLs	178
IPv4 ACL Switch Unsupported Features	178
Access List Numbers	178
Numbered Standard IPv4 ACLs	179
Numbered Extended IPv4 ACLs	180
Named IPv4 ACLs	180
ACL Logging	181
Hardware and Software Treatment of IP ACLs	181
VLAN Map Configuration Guidelines	182
VLAN Maps with Router ACLs	183
VLAN Maps and Router ACL Configuration Guidelines	183
Time Ranges for ACLs	184
IPv4 ACL Interface Considerations	184
How to Configure ACLs	185
Configuring IPv4 ACLs	185
Creating a Numbered Standard ACL	185
Creating a Numbered Extended ACL	187
Creating Named Standard ACLs	190
Creating Extended Named ACLs	191
Configuring Time Ranges for ACLs	193
Applying an IPv4 ACL to a Terminal Line	194
Applying an IPv4 ACL to an Interface	196
Creating Named MAC Extended ACLs	197
Applying a MAC ACL to a Layer 2 Interface	198
Configuring VLAN Maps	200
Creating a VLAN Map	201
Applying a VLAN Map to a VLAN	203

Monitoring IPv4 ACLs	204
Configuration Examples for ACLs	205
Examples: Using Time Ranges with ACLs	205
Examples: Including Comments in ACLs	205
IPv4 ACL Configuration Examples	206
ACLs in a Small Networked Office	206
Examples: ACLs in a Small Networked Office	207
Example: Numbered ACLs	208
Examples: Extended ACLs	208
Examples: Named ACLs	209
Examples: Time Range Applied to an IP ACL	209
Examples: Configuring Commented IP ACL Entries	210
Examples: ACL Logging	210
Configuration Examples for ACLs and VLAN Maps	212
Example: Creating an ACL and a VLAN Map to Deny a Packet	212
Example: Creating an ACL and a VLAN Map to Permit a Packet	212
Example: Default Action of Dropping IP Packets and Forwarding MAC Packets	212
Example: Default Action of Dropping MAC Packets and Forwarding IP Packets	213
Example: Default Action of Dropping All Packets	213
Configuration Examples for Using VLAN Maps in Your Network	214
Example: Wiring Closet Configuration	214
Example: Restricting Access to a Server on Another VLAN	215
Example: Denying Access to a Server on Another VLAN	215
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs	216
Example: ACLs and Switched Packets	216
Example: ACLs and Bridged Packets	217
Example: ACLs and Routed Packets	217
Example: ACLs and Multicast Packets	218
<hr/>	
CHAPTER 12	Configuring IPv6 ACLs 219
	Finding Feature Information 219
	IPv6 ACLs Overview 219
	Switch Stacks and IPv6 ACLs 220
	ACL Precedence 220

VLAN Maps	221
Interactions with Other Features and Switches	221
Restrictions for IPv6 ACLs	221
Default Configuration for IPv6 ACLs	222
Configuring IPv6 ACLs	222
Attaching an IPv6 ACL to an Interface	226
Configuring a VLAN Map	227
Applying a VLAN Map to a VLAN	229
Monitoring IPv6 ACLs	230

CHAPTER 13**Configuring DHCP 233**

Finding Feature Information	233
Information About DHCP	233
DHCP Server	233
DHCP Relay Agent	233
DHCP Snooping	234
Option-82 Data Insertion	235
Cisco IOS DHCP Server Database	238
DHCP Snooping Binding Database	238
DHCP Snooping and Switch Stacks	239
How to Configure DHCP Features	240
Default DHCP Snooping Configuration	240
DHCP Snooping Configuration Guidelines	241
Configuring the DHCP Server	241
DHCP Server and Switch Stacks	241
Configuring the DHCP Relay Agent	241
Specifying the Packet Forwarding Address	242
Prerequisites for Configuring DHCP Snooping and Option 82	244
Enabling the Cisco IOS DHCP Server Database	245
Monitoring DHCP Snooping Information	245
Configuring DHCP Server Port-Based Address Allocation	246
Information About Configuring DHCP Server Port-Based Address Allocation	246
Default Port-Based Address Allocation Configuration	246
Port-Based Address Allocation Configuration Guidelines	247

Enabling the DHCP Snooping Binding Database Agent	247
Enabling DHCP Server Port-Based Address Allocation	249
Monitoring DHCP Server Port-Based Address Allocation	250

CHAPTER 14**Configuring IP Source Guard 251**

Finding Feature Information	251
Information About IP Source Guard	251
IP Source Guard	251
IP Source Guard for Static Hosts	252
IP Source Guard Configuration Guidelines	252
How to Configure IP Source Guard	253
Enabling IP Source Guard	253
Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port	255
Monitoring IP Source Guard	256
Additional References	257

CHAPTER 15**Configuring Dynamic ARP Inspection 259**

Finding Feature Information	259
Restrictions for Dynamic ARP Inspection	259
Understanding Dynamic ARP Inspection	261
Interface Trust States and Network Security	262
Rate Limiting of ARP Packets	263
Relative Priority of ARP ACLs and DHCP Snooping Entries	264
Logging of Dropped Packets	264
Default Dynamic ARP Inspection Configuration	264
Relative Priority of ARP ACLs and DHCP Snooping Entries	265
Configuring ARP ACLs for Non-DHCP Environments	265
Configuring Dynamic ARP Inspection in DHCP Environments	267
Limiting the Rate of Incoming ARP Packets	270
Performing Dynamic ARP Inspection Validation Checks	271
Monitoring DAI	273
Verifying the DAI Configuration	273
Additional References	274

CHAPTER 16

Configuring IPv6 First Hop Security 275

- Finding Feature Information **275**
- Prerequisites for First Hop Security in IPv6 **275**
- Restrictions for First Hop Security in IPv6 **276**
- Information about First Hop Security in IPv6 **276**
- How to Configure an IPv6 Snooping Policy **278**
- How to Attach an IPv6 Snooping Policy to an Interface **280**
- How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface **282**
- How to Attach an IPv6 Snooping Policy to VLANs Globally **283**
- How to Configure the IPv6 Binding Table Content **283**
- How to Configure an IPv6 Neighbor Discovery Inspection Policy **284**
 - How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface **286**
 - How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface **287**
 - How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally **288**
- How to Configure an IPv6 Router Advertisement Guard Policy **288**
 - How to Attach an IPv6 Router Advertisement Guard Policy to an Interface **291**
 - How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface **292**
 - How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally **293**
- How to Configure an IPv6 DHCP Guard Policy **293**
 - How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface **295**
 - How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface **296**
 - How to Attach an IPv6 DHCP Guard Policy to VLANs Globally **297**
- How to Configure IPv6 Source Guard **298**
 - How to Attach an IPv6 Source Guard Policy to an Interface **299**
 - How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface **300**
- How to Configure IPv6 Prefix Guard **301**
 - How to Attach an IPv6 Prefix Guard Policy to an Interface **302**
 - How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface **302**
- Configuration Examples for IPv6 First Hop Security **303**
 - Examples: How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface **303**
 - Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface **303**

CHAPTER 17**Configuring SISF-Based Device Tracking 305**

- Information About SISF-Based Device Tracking 305
 - Overview of SISF-Based Device Tracking 305
 - Options to Enable SISF-Based Device Tracking 306
 - Migrating from Legacy Commands to SISF-Based Device-Tracking Commands 307
 - Migrating from Legacy IPDT and IPv6 Snooping to SISF-Based Device Tracking 307
 - IPDT, IPv6 Snooping, and SISF-Based Device Tracking CLI Compatibility 308
- How to Configure SISF-Based Device Tracking 311
 - Manually Enabling SISF-Based Device Tracking 311
 - Applying the Default Device Tracking Policy to a Target 311
 - Creating a Custom Device Tracking Policy with Custom Settings 312
 - Attaching a Device Tracking Policy to an Interface 315
 - Attaching a Device Tracking Policy to a VLAN 316
 - Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE Everest 16.5.x 317
 - Configuring a Multi-Switch Network to Stop Creating Binding Entries from a Trunk Port 317
- Configuration Examples for SISF-Based Device Tracking 319
 - Example: Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE Everest 16.5.x 319
 - Example: Disabling IPv6 Device Tracking on a Target 320
 - Example: Enabling IPv6 for SVI on VLAN (To Mitigate the Duplicate Address Problem) 320
 - Example: Mitigating the IPv4 Duplicate Address Problem 320
 - Example: Avoiding a Short Device-Tracking Binding Reachable Time 322
- Feature History and Information for SISF-Based Device Tracking 322

CHAPTER 18**Configuring IEEE 802.1x Port-Based Authentication 325**

- Finding Feature Information 325
- Information About 802.1x Port-Based Authentication 325
 - Port-Based Authentication Process 326
 - Port-Based Authentication Initiation and Message Exchange 328
- Authentication Manager for Port-Based Authentication 330
 - Port-Based Authentication Methods 330
 - Per-User ACLs and Filter-Ids 330
 - Port-Based Authentication Manager CLI Commands 331

Ports in Authorized and Unauthorized States	332
Port-Based Authentication and Switch Stacks	333
802.1x Host Mode	334
802.1x Multiple Authentication Mode	334
Multi-auth Per User VLAN assignment	335
MAC Move	336
MAC Replace	337
802.1x Accounting	337
802.1x Accounting Attribute-Value Pairs	338
802.1x Readiness Check	339
Switch-to-RADIUS-Server Communication	339
802.1x Authentication with VLAN Assignment	339
802.1x Authentication with Per-User ACLs	341
802.1x Authentication with Downloadable ACLs and Redirect URLs	342
Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL	342
Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs	343
VLAN ID-based MAC Authentication	343
802.1x Authentication with Guest VLAN	344
802.1x Authentication with Restricted VLAN	345
802.1x Authentication with Inaccessible Authentication Bypass	345
Inaccessible Authentication Bypass Support on Multiple-Authentication Ports	346
Inaccessible Authentication Bypass Authentication Results	346
Inaccessible Authentication Bypass Feature Interactions	346
802.1x Critical Voice VLAN	347
802.1x User Distribution	348
802.1x User Distribution Configuration Guidelines	348
IEEE 802.1x Authentication with Voice VLAN Ports	348
IEEE 802.1x Authentication with Port Security	349
IEEE 802.1x Authentication with Wake-on-LAN	349
IEEE 802.1x Authentication with MAC Authentication Bypass	350
Network Admission Control Layer 2 IEEE 802.1x Validation	351
Flexible Authentication Ordering	352
OpenIx Authentication	352
Multidomain Authentication	353

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)	354
Voice Aware 802.1x Security	355
Common Session ID	356
How to Configure 802.1x Port-Based Authentication	356
Default 802.1x Authentication Configuration	356
802.1x Authentication Configuration Guidelines	358
802.1x Authentication	358
VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass	359
MAC Authentication Bypass	359
Maximum Number of Allowed Devices Per Port	360
Configuring 802.1x Readiness Check	360
Configuring Voice Aware 802.1x Security	362
Configuring 802.1x Violation Modes	363
Configuring 802.1x Authentication	365
Configuring 802.1x Port-Based Authentication	365
Configuring the Switch-to-RADIUS-Server Communication	368
Configuring the Host Mode	369
Configuring Periodic Re-Authentication	370
Changing the Quiet Period	372
Changing the Switch-to-Client Retransmission Time	373
Setting the Switch-to-Client Frame-Retransmission Number	374
Setting the Re-Authentication Number	375
Enabling MAC Move	376
Enabling MAC Replace	377
Configuring 802.1x Accounting	378
Configuring a Guest VLAN	380
Configuring a Restricted VLAN	381
Configuring Number of Authentication Attempts on a Restricted VLAN	382
Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN	383
Example of Configuring Inaccessible Authentication Bypass	386
Configuring 802.1x Authentication with WoL	386
Configuring MAC Authentication Bypass	388
Configuring 802.1x User Distribution	388

Example of Configuring VLAN Groups	389
Configuring NAC Layer 2 802.1x Validation	390
Configuring an Authenticator Switch with NEAT	392
Configuring a Supplicant Switch with NEAT	393
Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs	396
Configuring Downloadable ACLs	396
Configuring a Downloadable Policy	397
Configuring VLAN ID-based MAC Authentication	399
Configuring Flexible Authentication Ordering	400
Configuring Open1x	401
Disabling 802.1x Authentication on the Port	402
Resetting the 802.1x Authentication Configuration to the Default Values	403
Monitoring 802.1x Statistics and Status	404

CHAPTER 19**Configuring Device Sensor 405**

About Device Sensor	405
MSP-IOS Sensor Device Classifier Interaction	406
Configuring Device Sensor	407
Enabling MSP	407
Enabling Accounting Augmentation	408
Creating a Cisco Discovery Protocol Filter	408
Creating an LLDP Filter	409
Creating a DHCP Filter	409
Applying a Protocol Filter to the Device Sensor Output	410
Tracking TLV Changes	411
Verifying the Device Sensor Configuration	412
Troubleshooting Commands	412
Restrictions for Device Sensor	412
Configuration Examples for the Device Sensor Feature	412
Feature Information for Device Sensor	413

CHAPTER 20**Web-Based Authentication 415**

Finding Feature Information	415
Web-Based Authentication Overview	415

Device Roles	416
Host Detection	417
Session Creation	417
Authentication Process	418
Local Web Authentication Banner	418
Web Authentication Customizable Web Pages	421
Guidelines	421
Authentication Proxy Web Page Guidelines	422
Redirection URL for Successful Login Guidelines	423
Web-based Authentication Interactions with Other Features	423
Port Security	423
LAN Port IP	423
Gateway IP	423
ACLs	423
Context-Based Access Control	424
EtherChannel	424
How to Configure Web-Based Authentication	424
Default Web-Based Authentication Configuration	424
Web-Based Authentication Configuration Guidelines and Restrictions	424
Configuring the Authentication Rule and Interfaces	426
Configuring AAA Authentication	427
Configuring Switch-to-RADIUS-Server Communication	430
Configuring the HTTP Server	431
Customizing the Authentication Proxy Web Pages	432
Specifying a Redirection URL for Successful Login	433
Configuring the Web-Based Authentication Parameters	434
Configuring a Web-Based Authentication Local Banner	435
Removing Web-Based Authentication Cache Entries	436
Monitoring Web-Based Authentication Status	437
<hr/>	
CHAPTER 21	Configuring Port-Based Traffic Control 439
	Overview of Port-Based Traffic Control 439
	Finding Feature Information 439
	Information About Storm Control 439

Storm Control	439
How Traffic Activity is Measured	440
Traffic Patterns	440
How to Configure Storm Control	441
Configuring Storm Control and Threshold Levels	441
Information About Protected Ports	444
Protected Ports	444
Default Protected Port Configuration	444
Protected Ports Guidelines	444
How to Configure Protected Ports	444
Configuring a Protected Port	444
Monitoring Protected Ports	446
Information About Port Blocking	446
Port Blocking	446
How to Configure Port Blocking	446
Blocking Flooded Traffic on an Interface	446
Monitoring Port Blocking	448
Prerequisites for Port Security	448
Restrictions for Port Security	448
Information About Port Security	448
Port Security	448
Types of Secure MAC Addresses	448
Sticky Secure MAC Addresses	449
Security Violations	449
Port Security Aging	450
Port Security and Switch Stacks	450
Default Port Security Configuration	451
Port Security Configuration Guidelines	451
Overview of Port-Based Traffic Control	452
How to Configure Port Security	453
Monitoring Port Security	460
Configuration Examples for Port Security	460
Information About Protocol Storm Protection	461
Protocol Storm Protection	461

Default Protocol Storm Protection Configuration	462
How to Configure Protocol Storm Protection	462
Enabling Protocol Storm Protection	462
Monitoring Protocol Storm Protection	463
Additional References for Port-Based Traffic Control	463

CHAPTER 22**Configuring Cisco TrustSec 465**

Information about Cisco TrustSec	465
Finding Feature Information	465
Cisco TrustSec Features	466
Feature Information for Cisco TrustSec	468

CHAPTER 23**Configuring Control Plane Policing 469**

Restrictions for CoPP	469
Information About CoPP	470
CoPP Overview	470
System-Defined Aspects of CoPP	470
User-Configurable Aspects of CoPP	474
Upgrading or Downgrading the Software Version	475
Software Version Upgrades and CoPP	475
Software Version Downgrades and CoPP	476
How to Configure CoPP	477
Enabling a CPU Queue or Changing the Policer Rate	477
Disabling a CPU Queue	478
Setting the Default Policer Rates for All CPU Queues	479
Creating A User-Defined Class Map	480
Configuration Examples for CoPP	481
Example: Enabling a CPU Queue or Changing the Policer Rate of a CPU Queue	481
Example: Disabling a CPU Queue	482
Example: Setting the Default Policer Rates for All CPU Queues	483
Example: Creating a User-Defined Class Map	484
Monitoring CoPP	485
Feature History and Information for CoPP	486



CHAPTER 1

Preventing Unauthorized Access

- [Finding Feature Information](#), on page 1
- [Restrictions for Controlling Switch Access with Passwords and Privileges](#), on page 1
- [Information About Passwords and Privilege Levels](#), on page 2
- [How to Control Switch Access with Passwords and Privilege Levels](#), on page 4
- [Monitoring Switch Access](#), on page 14
- [Configuration Examples for Setting Passwords and Privilege Levels](#), on page 15
- [Additional References](#), on page 16

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

- Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Related Topics

- [Disabling Password Recovery](#), on page 7
- [Password Recovery](#), on page 2

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

This table shows the default password and privilege level configuration.

Table 1: Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Additional Password Security

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

Related Topics

[Protecting Enable and Enable Secret Passwords with Encryption](#), on page 5

[Example: Protecting Enable and Enable Secret Passwords with Encryption](#), on page 15

Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set

the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Related Topics

[Disabling Password Recovery](#), on page 7

[Restrictions for Controlling Switch Access with Passwords and Privileges](#), on page 1

Terminal Line Telnet Configuration

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it when you set a Telnet password for a terminal line.

Related Topics

[Setting a Telnet Password for a Terminal Line](#), on page 8

[Example: Setting a Telnet Password for a Terminal Line](#), on page 15

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Related Topics

[Configuring Username and Password Pairs](#), on page 10

Privilege Levels

Cisco devices use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

Privilege Levels on Lines

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the

higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

Command Privilege Levels

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

Related Topics

[Setting the Privilege Level for a Command](#), on page 11

[Example: Setting the Privilege Level for a Command](#), on page 15

[Changing the Default Privilege Level for Lines](#), on page 13

[Logging into and Exiting a Privilege Level](#), on page 14

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Follow these steps to set or change a static enable password:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	enable password <i>password</i> Example: Device(config)# enable password secret321	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined.

	Command or Action	Purpose
		<p>For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Ctrl-v when you create the password; for example, to create the password abc?123, do this:</p> <ol style="list-style-type: none"> a. Enter abc. b. Enter Ctrl-v. c. Enter ?123. <p>When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Example: Setting or Changing a Static Enable Password](#), on page 15

Protecting Enable and Enable Secret Passwords with Encryption

Follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • <code>enable password [level level] {password encryption-type encrypted-password}</code> • <code>enable secret [level level] {password encryption-type encrypted-password}</code> Example: Device(config)# enable password example102 or Device(config)# enable secret level 1 password secret123sample	<ul style="list-style-type: none"> • Defines a new password or changes an existing password for access to privileged EXEC mode. • Defines a secret password, which is saved using a nonreversible encryption method. <ul style="list-style-type: none"> • (Optional) For <i>level</i>, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For <i>encryption-type</i>, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. <p>Note If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.</p>

	Command or Action	Purpose
Step 4	service password-encryption Example: <pre>Device(config)# service password-encryption</pre>	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Additional Password Security](#), on page 2

[Example: Protecting Enable and Enable Secret Passwords with Encryption](#), on page 15

Disabling Password Recovery

Follow these steps to disable password recovery to protect the security of your switch:

Before you begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	system disable password recovery switch {all <1-9>} Example: Device(config)# system disable password recovery switch all	Disables password recovery. <ul style="list-style-type: none"> • <i>all</i> - Sets the configuration on switches in stack. • <1-9> - Sets the configuration on the Switch Number selected. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

What to do next

To remove **disable password recovery**, use the **no system disable password recovery switch all** global configuration command.

Related Topics

[Password Recovery](#), on page 2

[Restrictions for Controlling Switch Access with Passwords and Privileges](#), on page 1

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before you begin

- Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management port.
- The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Note If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	line vty 0 15 Example: <pre>Device(config)# line vty 0 15</pre>	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable Device. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 4	password <i>password</i> Example: <pre>Device(config-line)# password abcxyz543</pre>	Sets a Telnet password for the line or lines. For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Device(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Terminal Line Telnet Configuration](#), on page 3

[Example: Setting a Telnet Password for a Terminal Line](#), on page 15

Configuring Username and Password Pairs

Follow these steps to configure username and password pairs:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>username name [privilege level] {password encryption-type password}</p> <p>Example:</p> <pre>Device(config)# username adamsample privilege 1 password secret456</pre> <pre>Device(config)# username 111111111111 mac attribute</pre>	<p>Sets the username, privilege level, and password for each user.</p> <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word or the MAC address. Spaces and quotation marks are not allowed. • You can configure a maximum of 12000 clients each, for both username and MAC filter. • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For <i>password</i>, specify the password the user must enter to gain access to the Device. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.

	Command or Action	Purpose
Step 4	Use one of the following: <ul style="list-style-type: none"> • line console 0 • line vty 0 15 Example: <pre>Device(config)# line console 0</pre> or <pre>Device(config)# line vty 15</pre>	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).
Step 5	login local Example: <pre>Device(config-line)# login local</pre>	Enables local password checking at login time. Authentication is based on the username specified in Step 3.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Username and Password Pairs](#), on page 3

Setting the Privilege Level for a Command

Follow these steps to set the privilege level for a command:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	privilege mode level level command Example: Device(config)# privilege exec level 14 configure	Sets the privilege level for a command. <ul style="list-style-type: none"> • For <i>mode</i>, enter configure for global configuration mode, exec for EXEC mode, interface for interface configuration mode, or line for line configuration mode. • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password. • For <i>command</i>, specify the command to which you want to restrict access.
Step 4	enable password level level password Example: Device(config)# enable password level 14 SecretPswd14	Specifies the password to enable the privilege level. <ul style="list-style-type: none"> • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: Device# copy running-config	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	<code>startup-config</code>	

Related Topics

[Privilege Levels](#), on page 3

[Example: Setting the Privilege Level for a Command](#), on page 15

Changing the Default Privilege Level for Lines

Follow these steps to change the default privilege level for the specified line:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	line vty line Example: Device(config)# <code>line vty 10</code>	Selects the virtual terminal line on which to restrict access.
Step 4	privilege level level Example: Device(config)# <code>privilege level 15</code>	Changes the default privilege level for the line. For <i>level</i> , the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

What to do next

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

Related Topics

[Privilege Levels](#), on page 3

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

Procedure

	Command or Action	Purpose
Step 1	enable <i>level</i> Example: Device> <code>enable 15</code>	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For <i>level</i> , the range is 0 to 15.
Step 2	disable <i>level</i> Example: Device# <code>disable 1</code>	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For <i>level</i> , the range is 0 to 15.

Related Topics

[Privilege Levels](#), on page 3

Monitoring Switch Access

Table 2: Commands for Displaying DHCP Information

<code>show privilege</code>	Displays the privilege level configuration.
-----------------------------	---

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

This example shows how to change the enable password to *11u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Device(config)# enable password 11u2c3k4y5
```

Related Topics

[Setting or Changing a Static Enable Password](#), on page 4

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *\$1\$FaD0\$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Device(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Related Topics

[Protecting Enable and Enable Secret Passwords with Encryption](#), on page 5
[Additional Password Security](#), on page 2

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Device(config)# line vty 10  
Device(config-line)# password let45me67in89
```

Related Topics

[Setting a Telnet Password for a Terminal Line](#), on page 8
[Terminal Line Telnet Configuration](#), on page 3

Example: Setting the Privilege Level for a Command

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Device(config)# privilege exec level 14 configure  
Device(config)# enable password level 14 SecretPswd14
```

Related Topics

[Setting the Privilege Level for a Command](#), on page 11

[Privilege Levels](#), on page 3

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 2

Controlling Switch Access with Passwords and Privilege Levels

- [Finding Feature Information, on page 17](#)
- [Restrictions for Controlling Switch Access with Passwords and Privileges, on page 17](#)
- [Information About Passwords and Privilege Levels, on page 18](#)
- [How to Control Switch Access with Passwords and Privilege Levels, on page 20](#)
- [Monitoring Switch Access, on page 30](#)
- [Configuration Examples for Setting Passwords and Privilege Levels, on page 31](#)
- [Additional References, on page 32](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

- Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Related Topics

- [Disabling Password Recovery, on page 7](#)
- [Password Recovery, on page 2](#)

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

This table shows the default password and privilege level configuration.

Table 3: Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Additional Password Security

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

Related Topics

[Protecting Enable and Enable Secret Passwords with Encryption](#), on page 5

[Example: Protecting Enable and Enable Secret Passwords with Encryption](#), on page 15

Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set

the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

To re-enable password recovery, use the **service password-recovery** global configuration command.

Related Topics

[Disabling Password Recovery](#), on page 7

[Restrictions for Controlling Switch Access with Passwords and Privileges](#), on page 1

Terminal Line Telnet Configuration

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it when you set a Telnet password for a terminal line.

Related Topics

[Setting a Telnet Password for a Terminal Line](#), on page 8

[Example: Setting a Telnet Password for a Terminal Line](#), on page 15

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Related Topics

[Configuring Username and Password Pairs](#), on page 10

Privilege Levels

Cisco devices use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

Privilege Levels on Lines

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the

higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

Command Privilege Levels

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

Related Topics

[Setting the Privilege Level for a Command](#), on page 11

[Example: Setting the Privilege Level for a Command](#), on page 15

[Changing the Default Privilege Level for Lines](#), on page 13

[Logging into and Exiting a Privilege Level](#), on page 14

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Follow these steps to set or change a static enable password:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	enable password <i>password</i> Example: Device(config)# enable password secret321	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined.

	Command or Action	Purpose
		<p>For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Ctrl-v when you create the password; for example, to create the password abc?123, do this:</p> <ol style="list-style-type: none"> Enter abc. Enter Ctrl-v. Enter ?123. <p>When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Example: Setting or Changing a Static Enable Password](#), on page 15

Protecting Enable and Enable Secret Passwords with Encryption

Follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • <code>enable password [level level] {password encryption-type encrypted-password}</code> • <code>enable secret [level level] {password encryption-type encrypted-password}</code> Example: Device(config)# enable password example102 or Device(config)# enable secret level 1 password secret123sample	<ul style="list-style-type: none"> • Defines a new password or changes an existing password for access to privileged EXEC mode. • Defines a secret password, which is saved using a nonreversible encryption method. <ul style="list-style-type: none"> • (Optional) For <i>level</i>, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For <i>encryption-type</i>, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. <p>Note If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.</p>

	Command or Action	Purpose
Step 4	service password-encryption Example: <pre>Device(config)# service password-encryption</pre>	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Additional Password Security](#), on page 2

[Example: Protecting Enable and Enable Secret Passwords with Encryption](#), on page 15

Disabling Password Recovery

Follow these steps to disable password recovery to protect the security of your switch:

Before you begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	system disable password recovery switch {all <1-9>} Example: Device(config)# system disable password recovery switch all	Disables password recovery. <ul style="list-style-type: none"> • <i>all</i> - Sets the configuration on switches in stack. • <1-9> - Sets the configuration on the Switch Number selected. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

What to do next

To remove **disable password recovery**, use the **no system disable password recovery switch all** global configuration command.

Related Topics

[Password Recovery](#), on page 2

[Restrictions for Controlling Switch Access with Passwords and Privileges](#), on page 1

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before you begin

- Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management port.
- The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Note If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	line vty 0 15 Example: <pre>Device(config)# line vty 0 15</pre>	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable Device. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 4	password <i>password</i> Example: <pre>Device(config-line)# password abcxyz543</pre>	Sets a Telnet password for the line or lines. For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Device(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Terminal Line Telnet Configuration](#), on page 3

[Example: Setting a Telnet Password for a Terminal Line](#), on page 15

Configuring Username and Password Pairs

Follow these steps to configure username and password pairs:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>username name [privilege level] {password encryption-type password}</p> <p>Example:</p> <pre>Device(config)# username adamsample privilege 1 password secret456</pre> <pre>Device(config)# username 111111111111 mac attribute</pre>	<p>Sets the username, privilege level, and password for each user.</p> <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word or the MAC address. Spaces and quotation marks are not allowed. • You can configure a maximum of 12000 clients each, for both username and MAC filter. • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For <i>password</i>, specify the password the user must enter to gain access to the Device. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.

	Command or Action	Purpose
Step 4	Use one of the following: <ul style="list-style-type: none"> • line console 0 • line vty 0 15 Example: Device(config)# line console 0 or Device(config)# line vty 15	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).
Step 5	login local Example: Device(config-line)# login local	Enables local password checking at login time. Authentication is based on the username specified in Step 3.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Username and Password Pairs](#), on page 3

Setting the Privilege Level for a Command

Follow these steps to set the privilege level for a command:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>privilege mode level level command</p> <p>Example:</p> <pre>Device(config)# privilege exec level 14 configure</pre>	<p>Sets the privilege level for a command.</p> <ul style="list-style-type: none"> For <i>mode</i>, enter configure for global configuration mode, exec for EXEC mode, interface for interface configuration mode, or line for line configuration mode. For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password. For <i>command</i>, specify the command to which you want to restrict access.
Step 4	<p>enable password level level password</p> <p>Example:</p> <pre>Device(config)# enable password level 14 SecretPswd14</pre>	<p>Specifies the password to enable the privilege level.</p> <ul style="list-style-type: none"> For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config</pre>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	<code>startup-config</code>	

Related Topics

[Privilege Levels](#), on page 3

[Example: Setting the Privilege Level for a Command](#), on page 15

Changing the Default Privilege Level for Lines

Follow these steps to change the default privilege level for the specified line:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	line vty line Example: Device(config)# <code>line vty 10</code>	Selects the virtual terminal line on which to restrict access.
Step 4	privilege level level Example: Device(config)# <code>privilege level 15</code>	Changes the default privilege level for the line. For <i>level</i> , the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

What to do next

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

Related Topics

[Privilege Levels](#), on page 3

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

Procedure

	Command or Action	Purpose
Step 1	enable <i>level</i> Example: Device> <code>enable 15</code>	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For <i>level</i> , the range is 0 to 15.
Step 2	disable <i>level</i> Example: Device# <code>disable 1</code>	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For <i>level</i> , the range is 0 to 15.

Related Topics

[Privilege Levels](#), on page 3

Monitoring Switch Access

Table 4: Commands for Displaying DHCP Information

<code>show privilege</code>	Displays the privilege level configuration.
-----------------------------	---

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

This example shows how to change the enable password to *11u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Device(config)# enable password 11u2c3k4y5
```

Related Topics

[Setting or Changing a Static Enable Password](#), on page 4

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *\$1\$FaD0\$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Device(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Related Topics

[Protecting Enable and Enable Secret Passwords with Encryption](#), on page 5
[Additional Password Security](#), on page 2

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Device(config)# line vty 10  
Device(config-line)# password let45me67in89
```

Related Topics

[Setting a Telnet Password for a Terminal Line](#), on page 8
[Terminal Line Telnet Configuration](#), on page 3

Example: Setting the Privilege Level for a Command

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Device(config)# privilege exec level 14 configure  
Device(config)# enable password level 14 SecretPswd14
```

Related Topics

[Setting the Privilege Level for a Command](#), on page 11

[Privilege Levels](#), on page 3

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 3

Configuring TACACS+

- [Finding Feature Information, on page 33](#)
- [Prerequisites for TACACS+, on page 33](#)
- [Information About Controlling Switch Access with TACACS+, on page 34](#)
- [How to Configure Switch Access with TACACS+, on page 38](#)
- [Monitoring TACACS+, on page 45](#)
- [Additional References For Switch Access with TACACS+, on page 45](#)
- [Feature Information for Switch Access with TACACS+, on page 46](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for TACACS+

The following are the prerequisites for set up and configuration of switch access with TACACS+ (must be performed in the order presented):

1. Configure the switches with the TACACS+ server addresses.
2. Set an authentication key.
3. Configure the key from Step 2 on the TACACS+ servers.
4. Enable authentication, authorization, and accounting (AAA).
5. Create a login authentication method list.
6. Apply the list to the terminal lines.
7. Create an authorization and accounting method list.

The following are the prerequisites for controlling switch access with TACACS+:

- You must have access to a configured TACACS+ server to configure TACACS+ features on your switch. Also, you must have access to TACACS+ services maintained in a database on a TACACS+ daemon typically running on a LINUX or Windows workstation.
- We recommend a redundant connection between a switch stack and the TACACS+ server. This is to help ensure that the TACACS+ server remains accessible in case one of the connected stack members is removed from the switch stack.
- You need a system running the TACACS+ daemon software to use TACACS+ on your switch.
- To use TACACS+, it must be enabled.
- Authorization must be enabled on the switch to be used.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- To use any of the AAA commands listed in this section or elsewhere, you must first enable AAA with the **aaa new-model** command.
- At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting.
- The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.
- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Information About Controlling Switch Access with TACACS+

TACACS+ and Switch Access

This section describes TACACS+. TACACS+ provides detailed accounting information and flexible administrative control over the authentication and authorization processes. It is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

TACACS+ Overview

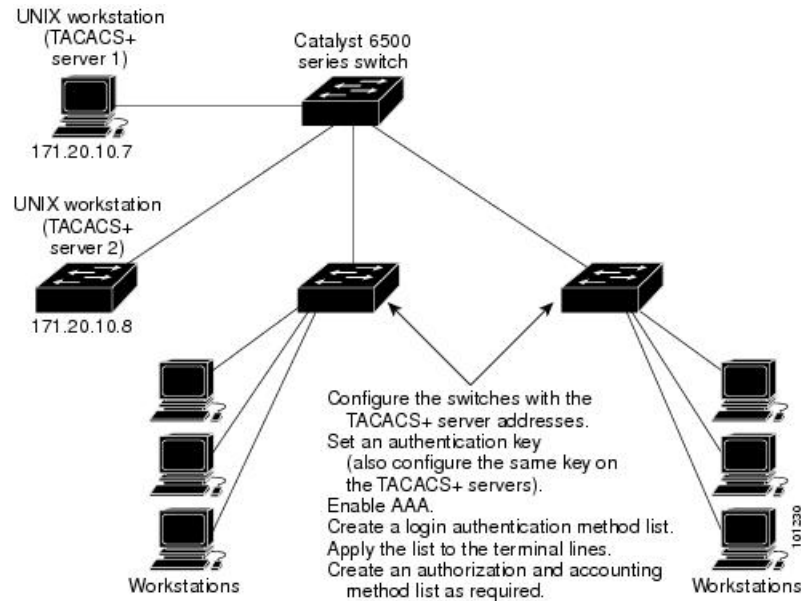
TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication,

authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers.

Figure 1: Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- **Authentication**—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name, service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- **Authorization**—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.
- **Accounting**—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:
 - **ACCEPT**—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
 - **REJECT**—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
 - **ERROR**—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an **ERROR** response is received, the switch typically tries to use an alternative method for authenticating the user.
 - **CONTINUE**—The user is prompted for additional authentication information.

After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an **ACCEPT** or **REJECT** authorization response. If an **ACCEPT** response is returned, the response contains data in the form of attributes that direct the **EXEC** or **NETWORK** session for that user and the services that the user can access:
 - Telnet, Secure Shell (SSH), rlogin, or privileged **EXEC** services
 - Connection parameters, including the host or client IP address, access list, and user timeouts

Method List

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

If a method list is configured under VTY lines, the corresponding method list must be added to AAA. The following example shows how to configure a method list under a VTY line:

```
Device# configure terminal
Device(config)# line vty 0 4
```

```
Device(config)# authorization commands 15 auth1
```

The following example shows how to configure a method list in AAA:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 auth1 group tacacs+
```

If no method list is configured under VTY lines, the default method list must be added to AAA. The following example shows a VTY configuration without a method list:

```
Device# configure terminal
Device(config)# line vty 0 4
```

The following example shows how to configure the default method list:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 default group tacacs+
```

TACACS+ Configuration Options

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

TACACS+ Login Authentication

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value

(AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.



Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

How to Configure Switch Access with TACACS+

This section describes how to configure your switch to support TACACS+.

Identifying the TACACS+ Server Host and Setting the Authentication Key

Follow these steps to identify the TACACS+ server host and set the authentication key:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	tacacs server <i>server-name</i> Example: Device(config)# tacacs server yourserver	Identifies the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. For <i>server-name</i> , specify the server name.
Step 4	address { ipv4 ipv6 } <i>ip address</i> Example:	Configures the IP address for the TACACS server.

	Command or Action	Purpose
	Device (config-server-tacacs) # address ipv4 10.0.1.12	
Step 5	exit Example: Device (config-server-tacacs) # exit	Exits the TACACS server mode and enters the global configuration mode.
Step 6	aaa new-model Example: Device (config) # aaa new-model	Enables AAA.
Step 7	aaa group server tacacs+ group-name Example: Device (config) # aaa group server tacacs+ your_server_group	(Optional) Defines the AAA server-group with a group name. This command puts the Device in a server group subconfiguration mode.
Step 8	server ip-address Example: Device (config) # server 10.1.2.3	(Optional) Associates a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 3.
Step 9	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 10	show running-config Example: Device# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Login Authentication

Follow these steps to configure TACACS+ login authentication:

Before you begin

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports.



Note To secure the for HTTP access by using AAA methods, you must configure the with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the for HTTP access by using AAA methods.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.
Step 4	aaa authentication login {default list-name} method1 [method2...] Example: Device(config)# aaa authentication login default tacacs+ local	Creates a login authentication method list. <ul style="list-style-type: none"> • To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For <i>list-name</i>, specify a character string to name the list you are creating.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • For <i>method1</i>..., specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. <p>Select one of these methods:</p> <ul style="list-style-type: none"> • <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. • <i>group tacacs+</i>—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. • <i>line</i> —Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. • <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username password global configuration command. • <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username name password global configuration command. • <i>none</i>—Do not use any authentication for login.
<p>Step 5</p>	<p>line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>]</p> <p>Example:</p> <pre>Device(config)# line 2 4</pre>	<p>Enters line configuration mode, and configures the lines to which you want to apply the authentication list.</p>
<p>Step 6</p>	<p>login authentication {default <i>list-name</i>}</p> <p>Example:</p>	<p>Applies the authentication list to a line or set of lines.</p>

	Command or Action	Purpose
	Device(config-line)# login authentication default	<ul style="list-style-type: none"> If you specify default, use the default list created with the aaa authentication login command. For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	end Example: Device(config-line)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	aaa authorization network tacacs+ Example: Device(config)# <code>aaa authorization network tacacs+</code>	Configures the switch for user TACACS+ authorization for all network-related service requests.
Step 4	aaa authorization exec tacacs+ Example: Device(config)# <code>aaa authorization exec tacacs+</code>	Configures the switch for user TACACS+ authorization if the user has privileged EXEC access. The exec keyword might return user profile information (such as autocommand information).
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Starting TACACS+ Accounting

Follow these steps to start TACACS+ Accounting:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa accounting network start-stop tacacs+ Example: Device(config)# aaa accounting network start-stop tacacs+	Enables TACACS+ accounting for all network-related service requests.
Step 4	aaa accounting exec start-stop tacacs+ Example: Device(config)# aaa accounting exec start-stop tacacs+	Enables TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To establish a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Establishing a Session with a Router if the AAA Server is Unreachable

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Monitoring TACACS+

Table 5: Commands for Displaying TACACS+ Information

Command	Purpose
show tacacs	Displays TACACS+ server statistics.

Additional References For Switch Access with TACACS+

Related Documents

Related Topic	Document Title
AAA configuration	Configuring Local Authentication and Authorization

MIBs

MIB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

Feature Information for Switch Access with TACACS+

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 6: Feature Information for Switch Access with TACACS+

Feature Name	Releases	Feature Information
Switch Access with TACACS+		TACACS+ provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through AAA and can be enabled only through AAA commands.



CHAPTER 4

Configuring RADIUS

- [Finding Feature Information, on page 47](#)
- [Prerequisites for Configuring RADIUS, on page 47](#)
- [Restrictions for Configuring RADIUS, on page 48](#)
- [Information about RADIUS, on page 49](#)
- [How to Configure RADIUS, on page 72](#)
- [Monitoring CoA Functionality, on page 88](#)
- [Additional References for Configuring Secure Shell, on page 88](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for Configuring RADIUS

This section lists the prerequisites for controlling Device access with RADIUS.

General:

- RADIUS and Authentication, Authorization, and Accounting (AAA) must be enabled to use any of the configuration commands in this chapter.
- RADIUS is facilitated through AAA and can be enabled only through AAA commands.
- Use the **aaa new-model** global configuration command to enable AAA.
- Use the **aaa authentication** global configuration command to define method lists for RADIUS authentication.
- Use **line** and **interface** commands to enable the defined method lists to be used.

- At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your Device.
- The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.
- To use the Change-of-Authorization (CoA) interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.
- A redundant connection between a switch stack and the RADIUS server is recommended. This is to help ensure that the RADIUS server remains accessible in case one of the connected stack members is removed from the switch stack.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Related Topics

[RADIUS and Switch Access](#), on page 49

[RADIUS Operation](#), on page 50

Restrictions for Configuring RADIUS

This topic covers restrictions for controlling Device access with RADIUS.

General:

- To prevent a lapse in security, you cannot configure RADIUS through a network management application.

RADIUS is not suitable in the following network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Related Topics

[RADIUS Overview](#), on page 49

Information about RADIUS

RADIUS and Switch Access

This section describes how to enable and configure RADIUS. RADIUS provides detailed accounting information and flexible administrative control over the authentication and authorization processes.

Related Topics

[Prerequisites for Configuring RADIUS](#), on page 47

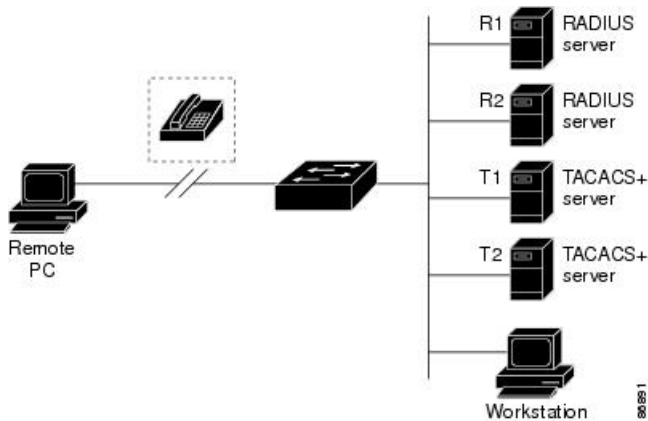
RADIUS Overview

RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.
- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validate users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco Device containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure: [Transitioning from RADIUS to TACACS+ Services](#) below.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see [Configuring IEEE 802.1x Port-Based Authentication](#) chapter.
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

Figure 2: Transitioning from RADIUS to TACACS+ Services



Related Topics

[Restrictions for Configuring RADIUS](#), on page 48

RADIUS Operation

When a user attempts to log in and authenticate to a Device that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
 - ACCEPT—The user is authenticated.
 - REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
 - CHALLENGE—A challenge requires additional data from the user.
 - CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Related Topics

[Prerequisites for Configuring RADIUS](#), on page 47

RADIUS Change of Authorization

The RADIUS Change of Authorization (CoA) provides a mechanism to change the attributes of an authentication, authorization, and accounting (AAA) session after it is authenticated. When a policy changes for a user or user group in AAA, administrators can send RADIUS CoA packets from the AAA server such as a Cisco Secure Access Control Server (ACS) to reinitialize authentication and apply the new policy. This section provides an overview of the RADIUS interface including available primitives and how they are used during a CoA.

- Change-of-Authorization Requests
- CoA Request Response Code
- CoA Request Commands
- Session Reauthentication
- Stacking Guidelines for Session Termination

A standard RADIUS interface is typically used in a pulled model where the request originates from a network attached device and the response come from the queried servers. Catalyst support the RADIUS CoA extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external AAA or policy servers.

The supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

This feature is integrated with Cisco Secure Access Control Server (ACS) 5.1.

The RADIUS interface is enabled by default on Catalyst . However, some basic configuration is required for the following attributes:

- Security and Password—refer to the “Preventing Unauthorized Access to Your Switch” section in this guide.
- Accounting—refer to the “Starting RADIUS Accounting” section in the Configuring Switch-Based Authentication chapter in this guide.

Cisco IOS software supports the RADIUS CoA extensions defined in RFC 5176 that are typically used in a push model to allow the dynamic reconfiguring of sessions from external AAA or policy servers. Per-session CoA requests are supported for session identification, session termination, host reauthentication, port shutdown, and port bounce. This model comprises one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA nonacknowledgement (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a AAA or policy server) and directed to the device that acts as a listener.

The table below shows the RADIUS CoA commands and vendor-specific attributes (VSAs) supported by Identity-Based Networking Services. All CoA commands must include the session identifier between the device and the CoA client.

Table 7: RADIUS CoA Commands Supported by Identity-Based Networking Services

CoA Command	Cisco VSA
Activate service	Cisco:Avpair="subscriber:command=activate-service" Cisco:Avpair="subscriber:service-name=<service-name>" Cisco:Avpair="subscriber:precedence=<precedence-number>" Cisco:Avpair="subscriber:activation-mode=replace-all"
Deactivate service	Cisco:Avpair="subscriber:command=deactivate-service" Cisco:Avpair="subscriber:service-name=<service-name>"
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"
Session query	Cisco:Avpair="subscriber:command=session-query"
Session reauthenticate	Cisco:Avpair="subscriber:command=reauthenticate" Cisco:Avpair="subscriber:reauthenticate-type=last" or Cisco:Avpair="subscriber:reauthenticate-type=rerun"
Session terminate	This is a standard disconnect request and does not require a VSA.
Interface template	Cisco:AVpair="interface-template-name=<interfacetemplate>"

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgment (ACK) [CoA-ACK]
- CoA non-acknowledgment (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination.

This table shows the IETF attributes are supported for this feature.

Table 8: Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

This table shows the possible values for the Error-Cause attribute.

Table 9: Error-Cause Values

Value	Explanation
201	Residual Session Context Removed
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension
407	Invalid Attribute Value
501	Administratively Prohibited
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable
507	Request Initiated
508	Multiple Session Selection Unsupported

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch.

The packet format for a CoA Request Response code as defined in RFC 5176 consists of the following fields: Code, Identifier, Length, Authenticator, and Attributes in the Type:Length:Value (TLV) format. The Attributes field is used to carry Cisco vendor-specific attributes (VSAs).

Related Topics

[CoA Request Commands](#), on page 55

Session Identification

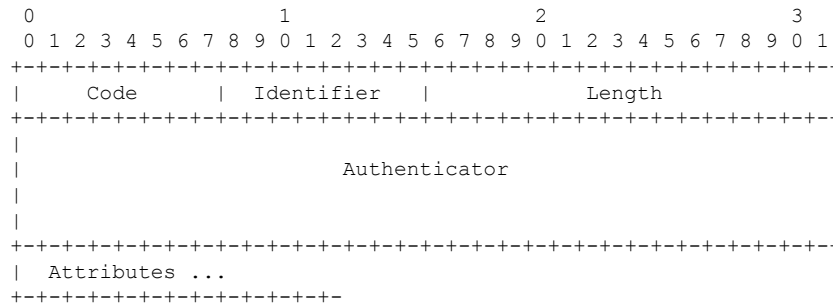
For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31 which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the “Invalid Attribute Value” error-code attribute.

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgment (NAK) or CoA-NAK with the error code “Invalid Attribute Value.”

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields: Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.



The attributes field is used to carry Cisco vendor-specific attributes (VSAs).

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

Related Topics

[CoA Disconnect-Request](#), on page 56

[CoA Request: Disable Host Port](#), on page 56

[CoA Request: Bounce-Port](#), on page 57

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgment (ACK) is sent. The attributes returned within CoA ACK will vary based on the CoA Request and are discussed in individual CoA Commands.

CoA NAK Response Code

A negative acknowledgment (NAK) indicates a failure to change the authorization state and can include attributes that indicate the reason for the failure. Use **show** commands to verify a successful CoA.

CoA Request Commands

Table 10: CoA Commands Supported on the

Command	Cisco VSA
¹	
Reauthenticate host	Cisco:Avpair="subscriber:command=reauthenticate"
Terminate session	This is a standard disconnect request that does not require a VSA.
Bounce host port	Cisco:Avpair="subscriber:command=bounce-host-port"
Disable host port	Cisco:Avpair="subscriber:command=disable-host-port"

¹ All CoA commands must include the session identifier between the and the CoA client.

Related Topics

[CoA Request Response Code](#), on page 54

Session Reauthentication

The AAA server typically generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message which contains a Cisco VSA in this form: *Cisco:Avpair="subscriber:command=reauthenticate"* and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by IEEE 802.1x, the switch responds by sending an EAPoL (Extensible Authentication Protocol over Lan) -RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized, or is authorized via guest VLAN, or critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

Session Reauthentication in a Switch Stack

When a switch stack receives a session reauthentication message:

- It checkpoints the need for a re-authentication before returning an acknowledgment (ACK).
- It initiates reauthentication for the appropriate session.
- If authentication completes with either success or failure, the signal that triggered the reauthentication is removed from the stack's member switch.
- If the stack's active switch fails before authentication completes, reauthentication is initiated after active switch changeover based on the original command (which is subsequently removed).
- If the active switch fails before sending an ACK, the new active switch treats the re-transmitted command as a new command.

Session Termination

There are three types of CoA requests that can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes re-initialization of the authenticator state machine for the specified host, but does not restrict that host access to the network.

To restrict a host's access to the network, use a CoA Request with the `Cisco:Avpair="subscriber:command=disable-host-port"` VSA. This command is useful when a host is known to be causing problems on the network, and you need to immediately block network access for the host. When you want to restore network access on the port, re-enable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs to acquire a new IP address (for example, after a VLAN change), terminate the session on the host port with port-bounce (temporarily disable and then re-enable the port).

CoA Disconnect-Request

This command is a standard Disconnect-Request. If the session cannot be located, the switch returns a Disconnect-NAK message with the "Session Context Not Found" error-code attribute. If the session is located, the switch terminates the session. After the session has been completely removed, the switch returns a Disconnect-ACK.

If the switch fails-over to a standby switch before returning a Disconnect-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the session is not found following re-sending, a Disconnect-ACK is sent with the "Session Context Not Found" error-code attribute.

Related Topics

[Session Identification](#), on page 54

CoA Request: Disable Host Port

The RADIUS server CoA disable port command administratively shuts down the authentication port that is hosting a session, resulting in session termination. This command is useful when a host is known to cause problems on the network and network access needs to be immediately blocked for the host. To restore network

access on the port, reenable it using a non-RADIUS mechanism. This command is carried in a standard CoA-Request message that has this new vendor-specific attribute (VSA):

```
Cisco:Avpair="subscriber:command=disable-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the “Session Identification” section. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.



Note A Disconnect-Request failure following command re-sending could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

Related Topics

[Session Identification](#), on page 54

CoA Request: Bounce-Port

A RADIUS server CoA bounce port sent from a RADIUS server can cause a link flap on an authentication port, which triggers DHCP renegotiation from one or more hosts connected to this port. This incident can occur when there is a VLAN change and the endpoint is a device (such as a printer) that does not have a mechanism to detect a change on this authentication port. The CoA bounce port is carried in a standard CoA-Request message that contains the following VSA:

```
Cisco:Avpair="subscriber:command=bounce-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, re-enables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is re-started on the new active switch.

Related Topics

[Session Identification](#), on page 54

Stacking Guidelines for Session Termination

No special handling is required for CoA Disconnect-Request messages in a switch stack.

Stacking Guidelines for CoA-Request Bounce-Port

Because the **bounce-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **bounce-port** command, it checkpoints the following information before returning a CoA-ACK message:

- the need for a port-bounce
- the port-id (found in the local session context)

The switch initiates a port-bounce (disables the port for 10 seconds, then re-enables it).

If the port-bounce is successful, the signal that triggered the port-bounce is removed from the standby switch.

If the active switch fails before the port-bounce completes, a port-bounce is initiated after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

Stacking Guidelines for CoA-Request Disable-Port

Because the **disable-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **disable-port** command, it verifies this information before returning a CoA-ACK message:

- the need for a port-disable
- the port-id (found in the local session context)

The switch attempts to disable the port.

If the port-disable operation is successful, the signal that triggered the port-disable is removed from the standby switch.

If the active switch fails before the port-disable operation completes, the port is disabled after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string

- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the %RADIUS-4-RADIUS_DEAD message appears, and then the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings.

Related Topics

[Identifying the RADIUS Server Host](#)

[Defining AAA Server Groups](#), on page 77

[Configuring Settings for All RADIUS Servers](#), on page 81

[Configuring RADIUS Login Authentication](#), on page 75

RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

Related Topics

[Configuring RADIUS Login Authentication](#), on page 75

AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to different UDP ports on a server at the same IP address. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one. If the first host entry fails to provide accounting services, the network access server tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order in which they are configured.)

Related Topics

[Defining AAA Server Groups](#), on page 77

AAA Authorization

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

Related Topics

[Configuring RADIUS Authorization for User Privileged Access and Network Services](#), on page 79

RADIUS Accounting

The AAA accounting feature tracks the services that users are using and the amount of network resources that they are consuming. When you enable AAA accounting, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. You can then analyze the data for network management, client billing, or auditing.

Related Topics

[Starting RADIUS Accounting](#), on page 80

Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attributevalue (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, the following AV pair causes Cisco’s “multiple named IP address pools” feature to be activated during IP authorization (during PPP’s Internet Protocol Control Protocol (IPCP) address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

If you insert an “*”, the AV pair “ip:addr-pool=first” becomes optional. Note that any AV pair can be made optional:

```
cisco-avpair= "ip:addr-pool*first"
```

The following example shows how to cause a user logging in from a network access server to have immediate access to EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

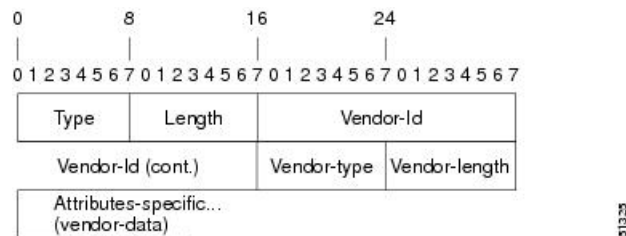
Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, “Remote Authentication Dial-In User Service (RADIUS).”

Attribute 26 contains the following three elements:

- Type
- Length
- String (also known as data)
 - Vendor-Id
 - Vendor-Type
 - Vendor-Length
 - Vendor-Data

The figure below shows the packet format for a VSA encapsulated “behind” attribute 26.

Figure 3: VSA Encapsulated Behind Attribute 26



Note It is up to the vendor to specify the format of their VSA. The Attribute-Specific field (also known as Vendor-Data) is dependent on the vendor's definition of that attribute.

The table below describes significant fields listed in the Vendor-Specific RADIUS IETF Attributes table (second table below), which lists supported vendor-specific RADIUS attributes (IETF attribute 26).

Table 11: Vendor-Specific Attributes Table Field Descriptions

Field	Description
Number	All attributes listed in the following table are extensions of IETF attribute 26.
Vendor-Specific Command Codes	A defined code used to identify a particular vendor. Code 9 defines Cisco VSAs, 311 defines Microsoft VSAs, and 529 defines Ascend VSAs.
Sub-Type Number	The attribute ID number. This number is much like the ID numbers of IETF attributes, except it is a “second layer” ID number encapsulated behind attribute 26.
Attribute	The ASCII string name of the attribute.
Description	Description of the attribute.

Table 12: Vendor-Specific RADIUS IETF Attributes

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
MS-CHAP Attributes				
26	311	1	MSCHAP-Response	Contains the response value provided by a PPP MS-CHAP user in response to the challenge. It is only used in Access-Request packets. This attribute is identical to the PPP CHAP Identifier. (RFC 2548
26	311	11	MSCHAP-Challenge	Contains the challenge sent by a network access server to an MS-CHAP user. It can be used in both Access-Request and Access-Challenge packets. (RFC 2548)
VPDN Attributes				
26	9	1	l2tp-cm-local-window-size	Specifies the maximum receive window size for L2TP control messages. This value is advertised to the peer during tunnel establishment.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	l2tp-drop-out-of-order	Respects sequence numbers on data packets by dropping those that are received out of order. This does not ensure that sequence numbers will be sent on data packets, just how to handle them if they are received.
26	9	1	l2tp-hello-interval	Specifies the number of seconds for the hello keepalive interval. Hello packets are sent when no data has been sent on a tunnel for the number of seconds configured here.
26	9	1	l2tp-hidden-avp	When enabled, sensitive AVPs in L2TP control messages are scrambled or hidden.
26	9	1	l2tp-nosession-timeout	Specifies the number of seconds that a tunnel will stay active with no sessions before timing out and shutting down.
26	9	1	tunnel-tos-reflect	Copies the IP ToS field from the IP header of each payload packet to the IP header of the tunnel packet for packets entering the tunnel at the LNS.
26	9	1	l2tp-tunnel-authen	If this attribute is set, it performs L2TP tunnel authentication.
26	9	1	l2tp-tunnel-password	Shared secret used for L2TP tunnel authentication and AVP hiding.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	l2tp-udp-checksum	This is an authorization attribute and defines whether L2TP should perform UDP checksums for data packets. Valid values are “yes” and “no.” The default is no.
Store and Forward Fax Attributes				
26	9	3	Fax-Account-Id-Origin	Indicates the account ID origin as defined by system administrator for the mmoip aaa receive-id or the mmoip aaa send-id commands.
26	9	4	Fax-Msg-Id=	Indicates a unique fax message identification number assigned by Store and Forward Fax.
26	9	5	Fax-Pages	Indicates the number of pages transmitted or received during this fax session. This page count includes cover pages.
26	9	6	Fax-Coverpage-Flag	Indicates whether or not a cover page was generated by the off-ramp gateway for this fax session. True indicates that a cover page was generated; false means that a cover page was not generated.
26	9	7	Fax-Modem-Time	Indicates the amount of time in seconds the modem sent fax data (x) and the amount of time in seconds of the total fax session (y), which includes both fax-mail and PSTN time, in the form x/y. For example, 10/15 means that the transfer time took 10 seconds, and the total fax session took 15 seconds.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	8	Fax-Connect-Speed	Indicates the modem speed at which this fax-mail was initially transmitted or received. Possible values are 1200, 4800, 9600, and 14400.
26	9	9	Fax-Recipient-Count	Indicates the number of recipients for this fax transmission. Until e-mail servers support Session mode, the number should be 1.
26	9	10	Fax-Process-Abort-Flag	Indicates that the fax session was cancelled or successful. True means that the session was cancelled; false means that the session was successful.
26	9	11	Fax-Dsn-Address	Indicates the address to which DSNs will be sent.
26	9	12	Fax-Dsn-Flag	Indicates whether or not DSN has been enabled. True indicates that DSN has been enabled; false means that DSN has not been enabled.
26	9	13	Fax-Mdn-Address	Indicates the address to which MDNs will be sent.
26	9	14	Fax-Mdn-Flag	Indicates whether or not message delivery notification (MDN) has been enabled. True indicates that MDN had been enabled; false means that MDN had not been enabled.
26	9	15	Fax-Auth-Status	Indicates whether or not authentication for this fax session was successful. Possible values for this field are success, failed, bypassed, or unknown.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	16	Email-Server-Address	Indicates the IP address of the e-mail server handling the on-ramp fax-mail message.
26	9	17	Email-Server-Ack-Flag	Indicates that the on-ramp gateway has received a positive acknowledgment from the e-mail server accepting the fax-mail message.
26	9	18	Gateway-Id	Indicates the name of the gateway that processed the fax session. The name appears in the following format: hostname.domain-name.
26	9	19	Call-Type	Describes the type of fax activity: fax receive or fax send.
26	9	20	Port-Used	Indicates the slot/port number of the Cisco AS5300 used to either transmit or receive this fax-mail.
26	9	21	Abort-Cause	If the fax session cancels, indicates the system component that signaled the cancel operation. Examples of system components that could trigger a cancel operation are FAP (Fax Application Process), TIFF (the TIFF reader or the TIFF writer), fax-mail client, fax-mail server, ESMTTP client, or ESMTTP server.
H323 Attributes				
26	9	23	Remote-Gateway-ID (h323-remote-address)	Indicates the IP address of the remote gateway.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	24	Connection-ID (h323-conf-id)	Identifies the conference ID.
26	9	25	Setup-Time (h323-setup-time)	Indicates the setup time for this connection in Coordinated Universal Time (UTC) formerly known as Greenwich Mean Time (GMT) and Zulu time.
26	9	26	Call-Origin (h323-call-origin)	Indicates the origin of the call relative to the gateway. Possible values are originating and terminating (answer).
26	9	27	Call-Type (h323-call-type)	Indicates call leg type. Possible values are telephony and VoIP .
26	9	28	Connect-Time (h323-connect-time)	Indicates the connection time for this call leg in UTC.
26	9	29	Disconnect-Time (h323-disconnect-time)	Indicates the time this call leg was disconnected in UTC.
26	9	30	Disconnect-Cause (h323-disconnect-cause)	Specifies the reason a connection was taken offline per Q.931 specification.
26	9	31	Voice-Quality (h323-voice-quality)	Specifies the impairment factor (ICPIF) affecting voice quality for a call.
26	9	33	Gateway-ID (h323-gw-id)	Indicates the name of the underlying gateway.
Large Scale Dialout Attributes				
26	9	1	callback-dialstring	Defines a dialing string to be used for callback.
26	9	1	data-service	No description available.
26	9	1	dial-number	Defines the number to dial.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	force-56	Determines whether the network access server uses only the 56 K portion of a channel, even when all 64 K appear to be available.
26	9	1	map-class	Allows the user profile to reference information configured in a map class of the same name on the network access server that dials out.
26	9	1	send-auth	Defines the protocol to use (PAP or CHAP) for username-password authentication following CLID authentication.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-name	<p>PPP name authentication. To apply for PAP, do not configure the ppp pap sent-name password command on the interface. For PAP, “preauth:send-name” and “preauth:send-secret” will be used as the PAP username and PAP password for outbound authentication. For CHAP, “preauth:send-name” will be used not only for outbound authentication, but also for inbound authentication. For a CHAP inbound case, the NAS will use the name defined in “preauth:send-name” in the challenge packet to the caller box.</p> <p>Note The send-name attribute has changed over time: Initially, it performed the functions now provided by both the send-name and remote-name attributes. Because the remote-name attribute has been added, the send-name attribute is restricted to its current behavior.</p>

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-secret	PPP password authentication. The vendor-specific attributes (VSAs) "preauth:send-name" and "preauth:send-secret" will be used as the PAP username and PAP password for outbound authentication. For a CHAP outbound case, both "preauth:send-name" and "preauth:send-secret" will be used in the response packet.
26	9	1	remote-name	Provides the name of the remote host for use in large-scale dial-out. Dialer checks that the large-scale dial-out remote name matches the authenticated name, to protect against accidental user RADIUS misconfiguration. (For example, dialing a valid phone number but connecting to the wrong device.)
Miscellaneous Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	2	Cisco-NAS-Port	<p>Specifies additional vendor specific attribute (VSA) information for NAS-Port accounting. To specify additional NAS-Port information in the form an Attribute-Value Pair (AVPair) string, use the radius-server vsa send global configuration command.</p> <p>Note This VSA is typically used in Accounting, but may also be used in Authentication (Access-Request) packets.</p>
26	9	1	min-links	Sets the minimum number of links for MLP.
26	9	1	proxyacl#<n>	Allows users to configure the downloadable user profiles (dynamic ACLs) by using the authentication proxy feature so that users can have the configured authorization to permit traffic going through the configured interfaces.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	spi	Carries the authentication information needed by the home agent to authenticate a mobile node during registration. The information is in the same syntax as the ip mobile secure host <addr> configuration command. Basically it contains the rest of the configuration command that follows that string, verbatim. It provides the Security Parameter Index (SPI), key, authentication algorithm, authentication mode, and replay protection timestamp range.

Related Topics

[Configuring the Device to Use Vendor-Specific RADIUS Attributes](#), on page 83

Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius server** global configuration commands.

Related Topics

[Configuring the Device for Vendor-Proprietary RADIUS Server Communication](#)

How to Configure RADIUS

Identifying the RADIUS Server Host

To apply these settings globally to all RADIUS servers communicating with the Device, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **key string**.

You can configure the Device to use AAA server groups to group existing server hosts for authentication. For more information, see Related Topics below.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the Device and the key string to be shared by both the server and the Device. For more information, see the RADIUS server documentation.

Follow these steps to configure per-server RADIUS server communication.

Before you begin

If you configure both global and per-server functions (timeout, retransmission, and key commands) on the device, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see Related Topics below.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius server <i>server name</i> Example: Device (config)# radius server rsim	
Step 4	address { ipv4 ipv6 } <i>ip address</i> { auth-port <i>port number</i> acct-port <i>port number</i> } Example: Device (config-radius-server)# address ipv4 124.2.2.12 auth-port 1612	(Optional) Specifies the RADIUS server parameters. For auth-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1645. The range is 0 to 65536. For acct-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1646.
Step 5	key string Example: Device (config-radius-server)# key rad123	(Optional) For key string , specify the authentication and encryption key used between the Device and the RADIUS daemon running on the RADIUS server.

	Command or Action	Purpose
		<p>Note The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the radius server command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.</p>
Step 6	<p>retransmit <i>value</i></p> <p>Example:</p> <pre>Device(config-radius-server)# retransmit 10</pre>	(Optional) Specifies the number of times a RADIUS request is resent when the server is not responding or responding slowly. The range is 1 to 100. This setting overrides the radius-server retransmit global configuration command setting.
Step 7	<p>timeout <i>seconds</i></p> <p>Example:</p> <pre>Device(config-radius-server)# timeout 60</pre>	(Optional) Specifies the time interval that the Device waits for the RADIUS server to reply before sending a request again. The range is 1 to 1000. This setting overrides the radius-server timeout global configuration command setting.
Step 8	<p>exit</p> <p>Example:</p> <pre>Device(config-server-tacacs)# exit</pre>	Exits the RADIUS server mode and enters the global configuration mode.
Step 9	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config</pre>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	<code>startup-config</code>	

Configuring RADIUS Login Authentication

Follow these steps to configure RADIUS login authentication:

Before you begin

To secure the device for HTTP access by using AAA methods, you must configure the device with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the device for HTTP access by using AAA methods.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa new-model Example: <pre>Device(config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authentication login {default list-name} method1 [method2...] Example: <pre>Device(config)# aaa authentication login default local</pre>	Creates a login authentication method list. <ul style="list-style-type: none"> • To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For <i>list-name</i>, specify a character string to name the list you are creating. • For <i>method1...</i>, specify the actual method the authentication algorithm tries. The additional methods of authentication are

	Command or Action	Purpose
		<p>used only if the previous method returns an error, not if it fails.</p> <p>Select one of these methods:</p> <ul style="list-style-type: none"> • <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. • <i>group radius</i>—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. • <i>line</i>—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. • <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username name password global configuration command. • <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username password global configuration command. • <i>none</i>—Do not use any authentication for login.
Step 5	<p>line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>]</p> <p>Example:</p> <pre>Device(config)# line 1 4</pre>	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 6	<p>login authentication {default <i>list-name</i>}</p> <p>Example:</p>	Applies the authentication list to a line or set of lines.

	Command or Action	Purpose
	<pre>Device(config)# login authentication default</pre>	<ul style="list-style-type: none"> If you specify default, use the default list created with the aaa authentication login command. For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	<pre>end</pre> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<pre>show running-config</pre> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<pre>copy running-config startup-config</pre> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[RADIUS Login Authentication](#), on page 59

[RADIUS Server Host](#), on page 58

Defining AAA Server Groups

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Follow these steps to define AAA server groups:

Procedure

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example:</p> <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius server name Example: Device(config)# radius server ISE	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode. The device also supports RADIUS for IPv6.
Step 4	address {ipv4 ipv6} {ip-address hostname} auth-port port-number acct-port port-number Example: Device(config-radius-server)# address ipv4 10.1.1.1 auth-port 1645 acct-port 1646	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 5	key string Example: Device(config-radius-server)# key cisco123	Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 6	end Example: Device(config-radius-server)# end	Exits RADIUS server configuration mode and returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Identifying the RADIUS Server Host](#)

[RADIUS Server Host](#), on page 58

[AAA Server Groups](#), on page 60

Configuring RADIUS Authorization for User Privileged Access and Network Services



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to configure RADIUS authorization for user privileged access and network services:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa authorization network radius Example: Device(config)# aaa authorization network radius	Configures the device for user RADIUS authorization for all network-related service requests.
Step 4	aaa authorization exec radius Example: Device(config)# aaa authorization exec radius	Configures the device for user RADIUS authorization if the user has privileged EXEC access. The exec keyword might return user profile information (such as autocommand information).
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Related Topics

[AAA Authorization](#), on page 60

Starting RADIUS Accounting

Follow these steps to start RADIUS accounting:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	aaa accounting network start-stop radius Example: <pre>Device(config)# aaa accounting network start-stop radius</pre>	Enables RADIUS accounting for all network-related service requests.
Step 4	aaa accounting exec start-stop radius Example: <pre>Device(config)# aaa accounting exec start-stop radius</pre>	Enables RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. This command guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Related Topics

[RADIUS Accounting](#), on page 60

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure settings for all RADIUS servers:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	radius-server key <i>string</i> Example: Device(config)# radius-server key your_server_key Device(config)# key your_server_key	Specifies the shared secret text string used between the switch and all RADIUS servers. Note The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 3	radius-server retransmit <i>retries</i> Example: Device(config)# radius-server retransmit 5	Specifies the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	radius-server timeout <i>seconds</i> Example: Device(config)# radius-server timeout 3	Specifies the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	radius-server deadtime <i>minutes</i> Example: Device(config)# radius-server deadtime 0	When a RADIUS server is not responding to authentication requests, this command specifies a time to stop the request on that server. This avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show running-config</code>	
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Related Topics

[Identifying the RADIUS Server Host RADIUS Server Host](#), on page 58

Configuring the Device to Use Vendor-Specific RADIUS Attributes

Follow these steps to configure the device to use vendor-specific RADIUS attributes:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	radius-server vsa send [accounting authentication] Example: Device(config)# <code>radius-server vsa send accounting</code>	Enables the device to recognize and use VSAs as defined by RADIUS IETF attribute 26. <ul style="list-style-type: none"> • (Optional) Use the accounting keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the authentication keyword to limit the set of recognized vendor-specific attributes to only authentication attributes.

	Command or Action	Purpose
		If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Vendor-Specific RADIUS Attributes](#), on page 60

Configuring the Device for Vendor-Proprietary RADIUS Server Communication

Follow these steps to configure the device to use vendor-proprietary RADIUS server communication:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius server <i>server name</i> Example:	Specifies the RADIUS server.

	Command or Action	Purpose
	Device (config) # radius server rsim	
Step 4	address { ipv4 ipv6 } ip address Example: Device (config-radius-server) # address ipv4 172.24.25.10	(Optional) Specifies the IP address of the RADIUS server.
Step 5	non-standard Example: Device (config-radius-server) # non-standard	Identifies that the RADIUS server using a vendor-proprietary implementation of RADIUS.
Step 6	key string Example: Device (config-radius-server) # key rad123	Specifies the shared secret text string used between the device and the vendor-proprietary RADIUS server. The device and the RADIUS server use this text string to encrypt passwords and exchange responses.
Step 7	exit Example: Device (config-radius-server) # exit	Exits the RADIUS server mode and enters the global configuration mode.
Step 8	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring CoA on the Device

Follow these steps to configure CoA on a device. This procedure is required.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device (config)# aaa new-model	Enables AAA.
Step 4	aaa server radius dynamic-author Example: Device (config)# aaa server radius dynamic-author	Configures the device as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 5	client <i>{ip-address name}</i> [vrf <i>vrfname</i>] [server-key string]	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
Step 6	server-key [0 7] <i>string</i> Example: Device (config-sg-radius)# server-key your_server_key	Configures the RADIUS key to be shared between a device and RADIUS clients.
Step 7	port <i>port-number</i> Example: Device (config-sg-radius)# port 25	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.

	Command or Action	Purpose
Step 8	auth-type {any all session-key} Example: Device(config-sg-radius)# auth-type any	Specifies the type of authorization the device uses for RADIUS clients. The client must match all the configured attributes for authorization.
Step 9	ignore session-key	(Optional) Configures the device to ignore the session-key. For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 10	ignore server-key Example: Device(config-sg-radius)# ignore server-key	(Optional) Configures the device to ignore the server-key. For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 11	authentication command bounce-port ignore Example: Device(config-sg-radius)# authentication command bounce-port ignore	(Optional) Configures the device to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
Step 12	authentication command disable-port ignore Example: Device(config-sg-radius)# authentication command disable-port ignore	(Optional) Configures the device to ignore a nonstandard command requesting that the port hosting a session be administratively shut down. Shutting down the port results in termination of the session. Use standard CLI or SNMP commands to re-enable the port.
Step 13	end Example: Device(config-sg-radius)# end	Returns to privileged EXEC mode.
Step 14	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 15	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring CoA Functionality

Table 13: Privileged EXEC show Commands

Command	Purpose
<code>show aaa attributes protocol radius</code>	Displays AAA attributes of RADIUS commands.

Table 14: Global Troubleshooting Commands

Command	Purpose
<code>debug radius</code>	Displays information for troubleshooting RADIUS.
<code>debug aaa coa</code>	Displays information for troubleshooting CoA processing.
<code>debug aaa pod</code>	Displays information for troubleshooting POD packets.
<code>debug aaa subsys</code>	Displays information for troubleshooting POD packets.
<code>debug cmdhd [detail error events]</code>	Displays information for troubleshooting command headers.

For detailed information about the fields in these displays, see the command reference for this release.

Additional References for Configuring Secure Shell

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 5

Configuring Kerberos

- [Finding Feature Information, on page 91](#)
- [Prerequisites for Controlling Switch Access with Kerberos, on page 91](#)
- [Information about Kerberos, on page 92](#)
- [How to Configure Kerberos, on page 95](#)
- [Monitoring the Kerberos Configuration, on page 95](#)
- [Additional References, on page 95](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for Controlling Switch Access with Kerberos

The following are the prerequisites for controlling switch access with Kerberos.

- So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.
- A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.

- The Kerberos realm name *must* be in all uppercase characters.

Information about Kerberos

This section provides Kerberos information.

Kerberos and Switch Access

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party.



Note In the Kerberos configuration examples, the trusted third party can be any switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Kerberos Overview

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited life span, are stored in user credential caches. The Kerberos server uses the tickets instead of user names and passwords to authenticate users and network services.



Note A Kerberos server can be any switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

Kerberos supports these network services:

- Telnet
- rlogin
- rsh

This table lists the common Kerberos-related terms and definitions.

Table 15: Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs ² and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default life span of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form <i>user@REALM</i> (for example, <i>smith@EXAMPLE.COM</i>). A Kerberos principal with a Kerberos instance has the form <i>user/instance@REALM</i> (for example, <i>smith/admin@EXAMPLE.COM</i>). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. Note The Kerberos principal and instance names <i>must</i> be in all lowercase characters. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
KDC ³	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
KEYTAB ⁴	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB ⁵ .
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. Note The Kerberos principal name <i>must</i> be in all lowercase characters.

Term	Definition
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

² ticket granting ticket

³ key distribution center

⁴ key table

⁵ server table

Kerberos Operation

A Kerberos server can be a device that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a device as a Kerberos server, remote users must follow these steps:

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
 - If the decryption is successful, the user is authenticated to the switch.
 - If the decryption is not successful, the user repeats Step 2 either by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the “Obtaining a TGT from a KDC” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the “Authenticating to Network Services” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

How to Configure Kerberos

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

Monitoring the Kerberos Configuration

To display the Kerberos configuration, use the following commands:

- **show running-config**
- **show kerberos creds**: Lists the credentials in a current user’s credentials cache.
- **clear kerberos creds**: Destroys all credentials in a current user’s credentials cache, including those forwarded.

Additional References

Related Documents

Related Topic	Document Title
Kerberos Commands	<i>Cisco IOS Security Command Reference</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 6

MACsec Encryption

- [Finding Feature Information, on page 97](#)
- [Prerequisites for MACsec Encryption, on page 97](#)
- [Information About MACsec Encryption, on page 98](#)
- [How to Configure MACsec Encryption, on page 103](#)
- [Configuration Examples for MACsec Encryption, on page 125](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for MACsec Encryption

Prerequisites for Certificate-Based MACsec

- Ensure that you have a Certificate Authority (CA) server configured for your network.
- Generate a CA certificate.
- Ensure that you have configured Cisco Identity Services Engine (ISE) Release 2.0.
- Ensure that both the participating devices, the CA server, and Cisco Identity Services Engine (ISE) are synchronized using Network Time Protocol (NTP). If time is not synchronized on all your devices, certificates will not be validated.
- Ensure that 802.1x authentication and AAA are configured on your device.

Information About MACsec Encryption

MACsec is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec-capable devices. These Catalyst switches support 802.1AE encryption with MACsec Key Agreement (MKA) on downlink ports for encryption between the switch and host device. The switch also supports MACsec encryption for switch-to-switch (inter-network device) security using both Cisco TrustSec Network Device Admission Control (NDAC), Security Association Protocol (SAP) and MKA-based key exchange protocol.

Link layer security can include both packet authentication between switches and MACsec encryption between switches (encryption is optional).



Note MACsec is not supported with the NPE license or the LAN Base service image.

Table 16: MACsec Support on Switch Ports

Interface	Connections	MACsec support
Uplink ports	Switch-to-switch	MACsec MKA encryption Cisco TrustSec NDAC MACsec

Cisco TrustSec and Cisco SAP are meant only for switch-to-switch links and are not supported on switch ports connected to end hosts, such as PCs or IP phones. MKA is supported on switch-to-host facing links (downlink) as well as switch-to-switch links (uplink). Host-facing links typically use flexible authentication ordering for handling heterogeneous devices with or without IEEE 802.1x, and can optionally use MKA-based MACsec encryption. Cisco NDAC and SAP are mutually exclusive with Network Edge Access Topology (NEAT), which is used for compact switches to extend security outside the wiring closet.

Recommendations for MACsec Encryption

This section lists the recommendations for configuring MACsec encryption:

- Use the confidentiality (encryption) offset as 0 in switch-to-host connections.
- Use Bidirectional Forwarding and Detection (BFD) timer value as 750 milliseconds for 10Gbps ports and 1.25 seconds for any port with speed above 10Gbps.
- Execute the **shutdown** command, and then the **no shutdown** command on a port, after changing any MKA policy or MACsec configuration for active sessions, so that the changes are applied to active sessions.
- Use Extended Packet Numbering (XPN) Cipher Suite for port speeds of 40Gbps and above.
- Set the connectivity association key (CAK) rekey overlap timer to 30 seconds or more.
- Do not use Cisco TrustSec Security Association Protocol (SAP) MACsec encryption for port speeds above 10Gbps.
- Do not enable both Cisco TrustSec SAP and uplink MKA at the same time on any interface.

Media Access Control Security and MACsec Key Agreement

MACsec, defined in 802.1AE, provides MAC-layer encryption over wired networks by using out-of-band methods for encryption keying. The MACsec Key Agreement (MKA) Protocol provides the required session keys and manages the required encryption keys. MKA and MACsec are implemented after successful authentication using the 802.1x Extensible Authentication Protocol (EAP-TLS) or Pre Shared Key (PSK) framework.

A switch using MACsec accepts either MACsec or non-MACsec frames, depending on the policy associated with the MKA peer. MACsec frames are encrypted and protected with an integrity check value (ICV). When the switch receives frames from the MKA peer, it decrypts them and calculates the correct ICV by using session keys provided by MKA. The switch compares that ICV to the ICV within the frame. If they are not identical, the frame is dropped. The switch also encrypts and adds an ICV to any frames sent over the secured port (the access point used to provide the secure MAC service to a MKA peer) using the current session key.

The MKA Protocol manages the encryption keys used by the underlying MACsec protocol. The basic requirements of MKA are defined in 802.1x-REV. The MKA Protocol extends 802.1x to allow peer discovery with confirmation of mutual authentication and sharing of MACsec secret keys to protect data exchanged by the peers.

The EAP framework implements MKA as a newly defined EAP-over-LAN (EAPOL) packet. EAP authentication produces a master session key (MSK) shared by both partners in the data exchange. Entering the EAP session ID generates a secure connectivity association key name (CKN). The switch acts as the authenticator for both uplink and downlink; and acts as the key server for downlink. It generates a random secure association key (SAK), which is sent to the client partner. The client is never a key server and can only interact with a single MKA entity, the key server. After key derivation and generation, the switch sends periodic transports to the partner at a default interval of 2 seconds.

The packet body in an EAPOL Protocol Data Unit (PDU) is referred to as a MACsec Key Agreement PDU (MKPDU). MKA sessions and participants are deleted when the MKA lifetime (6 seconds) passes with no MKPDU received from a participant. For example, if a MKA peer disconnects, the participant on the switch continues to operate MKA until 6 seconds have elapsed after the last MKPDU is received from the MKA peer.

MKA Policies

To enable MKA on an interface, a defined MKA policy should be applied to the interface. You can configure these options:

- Policy name, not to exceed 16 ASCII characters.
- Confidentiality (encryption) offset of 0, 30, or 50 bytes for each physical interface

Virtual Ports

Use virtual ports for multiple secured connectivity associations on a single physical port. Each connectivity association (pair) represents a virtual port. In uplink, you can have only one virtual port per physical port. In downlink, you can have a maximum of two virtual ports per physical port, of which one virtual port can be part of a data VLAN; the other must externally tag its packets for the voice VLAN. You cannot simultaneously host secured and unsecured sessions in the same VLAN on the same port. Because of this limitation, 802.1x multiple authentication mode is not supported.

The exception to this limitation is in multiple-host mode when the first MACsec supplicant is successfully authenticated and connected to a hub that is connected to the switch. A non-MACsec host connected to the

hub can send traffic without authentication because it is in multiple-host mode. We do not recommend using multi-host mode because after the first successful client, authentication is not required for other clients.

Virtual ports represent an arbitrary identifier for a connectivity association and have no meaning outside the MKA Protocol. A virtual port corresponds to a separate logical port ID. Valid port IDs for a virtual port are 0x0002 to 0xFFFF. Each virtual port receives a unique secure channel identifier (SCI) based on the MAC address of the physical interface concatenated with a 16-bit port ID.

MKA Statistics

Some MKA counters are aggregated globally, while others are updated both globally and per session. You can also obtain information about the status of MKA sessions. See [Example: Displaying MKA Information, on page 128](#) for further information.

Key Lifetime and Hitless Key Rollover

A MACsec key chain can have multiple pre-shared keys (PSK) each configured with a key id and an optional lifetime. A key lifetime specifies at which time the key expires. In the absence of a lifetime configuration, the default lifetime is unlimited. When a lifetime is configured, MKA rolls over to the next configured pre-shared key in the key chain after the lifetime is expired. Time zone of the key can be local or UTC. Default time zone is UTC.

You can Key rolls over to the next key within the same key chain by configuring a second key in the key chain and configuring a lifetime for the first key. When the lifetime of the first key expires, it automatically rolls over to the next key in the list. If the same key is configured on both sides of the link at the same time, then the key rollover is hitless, that is, key rolls over without traffic interruption.

On all participating devices, the MACsec key chain must be synchronised by using Network Time Protocol (NTP) and the same time zone must be used. If all the participating devices are not synchronized, the connectivity association key (CAK) rekey will not be initiated on all the devices at the same time.



Note The lifetime of the keys need to be overlapped in order to achieve hitless key rollover.

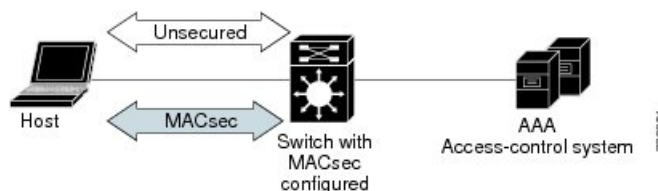
MACsec, MKA and 802.1x Host Modes

You can use MACsec and the MKA Protocol with 802.1x single-host mode, multi-host mode, or Multi Domain Authentication (MDA) mode. Multiple authentication mode is not supported.

Single-Host Mode

The figure shows how a single EAP authenticated session is secured by MACsec by using MKA

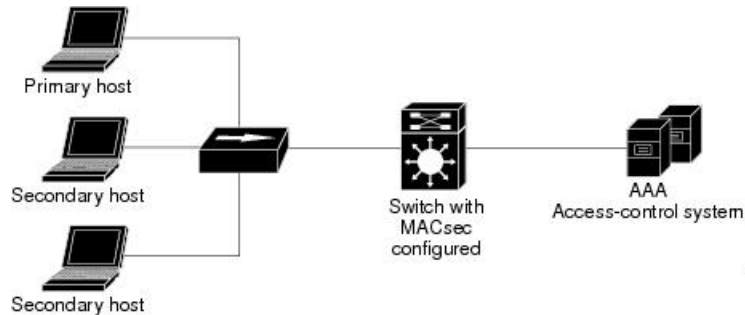
Figure 4: MACsec in Single-Host Mode with a Secured Data Session



Multiple Host Mode

In standard (not 802.1x REV) 802.1x multiple-host mode, a port is open or closed based on a single authentication. If one user, the primary secured client services client host, is authenticated, the same level of network access is provided to any host connected to the same port. If a secondary host is a MACsec supplicant, it cannot be authenticated and traffic would not flow. A secondary host that is a non-MACsec host can send traffic to the network without authentication because it is in multiple-host mode. The figure shows MACsec in Standard Multiple-Host Unsecured Mode.

Figure 5: MACsec in Multiple-Host Mode - Unsecured



Note Multi-host mode is not recommended because after the first successful client, authentication is not required for other clients, which is not secure.

In standard (not 802.1x REV) 802.1x multiple-domain mode, a port is open or closed based on a single authentication. If the primary user, a PC on data domain, is authenticated, the same level of network access is provided to any domain connected to the same port. If a secondary user is a MACsec supplicant, it cannot be authenticated and traffic would no flow. A secondary user, an IP phone on voice domain, that is a non-MACsec host, can send traffic to the network without authentication because it is in multiple-domain mode.

Understanding MACsec MKA using EAP-TLS

MACsec MKA is supported on switch-to-switch links. Using IEE 802.1X Port-based Authentication with Extensible Authentication Protocol (EAP-TLS), you can configure MACsec MKA between device uplink ports. EAP-TLS allows mutual authentication and obtains an MSK (master session key) from which the connectivity association key (CAK) is derived for MKA operations. Device certificates are carried, using EAP-TLS, for authentication to the AAA server.

Prerequisites for MACsec MKA using EAP-TLS

- Ensure that you have a Certificate Authority (CA) server configured for your network.
- Generate a CA certificate.
- Ensure that you have configured Cisco Identity Services Engine (ISE) Release 2.0.
- Ensure that both the participating devices, the CA server, and Cisco Identity Services Engine (ISE) are synchronized using Network Time Protocol (NTP). If time is not synchronized on all your devices, certificates will not be validated.

- Ensure that 802.1x authentication and AAA are configured on your device.

Limitations for MACsec MKA using EAP-TLS

- MKA is not supported on port-channels.
- MKA is not supported with High Availability and local authentication.
- MKA/EAPTLS is not supported for promiscuous PVLAN Primary port.
- While configuring MACsec MKA using EAP-TLS, MACsec secure channels encrypt counters does not increment before first Rekey.

Cisco TrustSec Overview

The table below lists the TrustSec features to be eventually implemented on TrustSec-enabled Cisco switches. Successive general availability releases of TrustSec will expand the number of switches supported and the number of TrustSec features supported per switch.

Cisco TrustSec Feature	Description
802.1AE Tagging (MACsec)	<p>Protocol for IEEE 802.1AE-based wire-rate hop-to-hop Layer 2 encryption.</p> <p>Between MACsec-capable devices, packets are encrypted on egress from the transmitting device, decrypted on ingress to the receiving device, and in the clear within the devices.</p> <p>This feature is only available between TrustSec hardware-capable devices.</p>
Endpoint Admission Control (EAC)	<p>EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth).</p>
Network Device Admission Control (NDAC)	<p>NDAC is an authentication process where each network device in the TrustSec domain can verify the credentials and trustworthiness of its peer device. NDAC utilizes an authentication framework based on IEEE 802.1X port-based authentication and uses EAP-FAST as its EAP method. Successful authentication and authorization in NDAC process results in Security Association Protocol negotiation for IEEE 802.1AE encryption.</p>

Cisco TrustSec Feature	Description
Security Association Protocol (SAP)	After NDAC authentication, the Security Association Protocol (SAP) automatically negotiates keys and the cipher suite for subsequent MACSec link encryption between TrustSec peers. SAP is defined in IEEE 802.11i.
Security Group Tag (SGT)	An SGT is a 16-bit single label indicating the security classification of a source in the TrustSec domain. It is appended to an Ethernet frame or an IP packet.
SGT Exchange Protocol (SXP)	Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.

When both ends of a link support 802.1AE MACsec, SAP negotiation occurs. An EAPOL-key exchange occurs between the supplicant and the authenticator to negotiate a cipher suite, exchange security parameters, and manage keys. Successful completion of these tasks results in the establishment of a security association (SA).

Depending on your software version and licensing and link hardware support, SAP negotiation can use one of these modes of operation:

- Galois Counter Mode (GCM)—authentication and encryption
- GCM authentication (GMAC)— GCM authentication, no encryption
- No Encapsulation—no encapsulation (clear text)
- Null—encapsulation, no authentication or encryption

How to Configure MACsec Encryption

Configuring MKA and MACsec

Default MACsec MKA Configuration

MACsec is disabled. No MKA policies are configured.

Configuring an MKA Policy

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	mka policy <i>policy name</i>	Identify an MKA policy, and enter MKA policy configuration mode. The maximum policy name length is 16 characters. Note The default MACsec cipher suite in the MKA policy will always be "GCM-AES-128". If the device supports both "GCM-AES-128" and "GCM-AES-256" ciphers, it is highly recommended to define and use a user defined MKA policy to include both 128 and 256 bits ciphers or only 256 bits cipher, as may be required.
Step 3	key-server <i>priority</i>	Configure MKA key server options and set priority (between 0-255). Note When value of key server priority is set to 255, the peer can not become the key server.
Step 4	macsec-cipher-suite <i>gcm-aes-128</i>	Configures cipher suite for deriving SAK with 128-bit encryption.
Step 5	confidentiality-offset <i>Offset value</i>	Set the Confidentiality (encryption) offset for each physical interface Note Offset Value can be 0, 30 or 50. If you are using Anyconnect on the client, it is recommended to use Offset 0.
Step 6	end	Returns to privileged EXEC mode.
Step 7	show mka policy	Verify your entries.

Example

This example configures the MKA policy:

```
Switch(config)# mka policy mka_policy
Switch(config-mka-policy)# key-server priority 200
Switch(config-mka-policy)# macsec-cipher-suite gcm-aes-128
```

```
Switch(config-mka-policy) # confidentiality-offset 30
Switch(config-mka-policy) # end
```

Configuring Switch-to-host MACsec Encryption

Follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch>enable	Enables privileged EXEC mode. Enter the password if prompted.
Step 2	configureterminal Example: Switch>configure terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i>	Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	switchport access vlan <i>vlan-id</i>	Configure the access VLAN for the port.
Step 5	switchport mode access	Configure the interface as an access port.
Step 6	macsec	Enable 802.1ae MACsec on the interface. The macsec command enables MKA MACsec on switch-to-host links (downlink ports) only.
Step 7	authentication event linksec fail action authorize vlan <i>vlan-id</i>	(Optional) Specify that the switch processes authentication link-security failures resulting from unrecognized user credentials by authorizing a restricted VLAN on the port after a failed authentication attempt.
Step 8	authentication host-mode multi-domain	Configure authentication manager mode on the port to allow both a host and a voice device to be authenticated on the 802.1x-authorized port. If not configured, the default host mode is single.
Step 9	authentication linksec policy must-secure	Set the LinkSec security policy to secure the session with MACsec if the peer is available. If not set, the default is <i>should secure</i> .
Step 10	authentication port-control auto	Enable 802.1x authentication on the port. The port changes to the authorized or unauthorized state based on the authentication exchange between the switch and the client.

	Command or Action	Purpose
Step 11	authentication periodic	Enable or Disable Reauthentication for this port .
Step 12	authentication timer reauthenticate	Enter a value between 1 and 65535 (in seconds). Obtains re-authentication timeout value from the server. Default re-authentication time is 3600 seconds.
Step 13	authentication violation protect	Configure the port to drop unexpected incoming MAC addresses when a new device connects to a port or when a device connects to a port after the maximum number of devices are connected to that port. If not configured, the default is to shut down the port.
Step 14	mka policy <i>policy name</i>	Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering the mka policy global configuration command).
Step 15	dot1x pae authenticator	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 16	spanning-tree portfast	Enable spanning tree Port Fast on the interface in all its associated VLANs. When Port Fast feature is enabled, the interface changes directly from a blocking state to a forwarding state without making the intermediate spanning-tree state changes
Step 17	end Example: <code>Switch(config)#end</code>	Returns to privileged EXEC mode.
Step 18	show authentication session interface <i>interface-id</i>	Verify the authorized session security status.
Step 19	show authentication session interface <i>interface-id</i> details	Verify the details of the security status of the authorized session.
Step 20	show macsec interface <i>interface-id</i>	Verify MacSec status on the interface.
Step 21	show mka sessions	Verify the established mka sessions.
Step 22	copy running-config startup-config Example: <code>Switch#copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MACsec MKA using PSK

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	key chain <i>key-chain-name</i> macsec	Configures a key chain and enters the key chain configuration mode.
Step 3	key <i>hex-string</i>	Configures a unique identifier for each key in the keychain and enters the keychain's key configuration mode. Note For 128-bit encryption, use 32 hex digit key-string. For 256-bit encryption, use 64 hex digit key-string.
Step 4	cryptographic-algorithm {<i>gcm-aes-128</i> / <i>gcm-aes-256</i>}	Set cryptographic authentication algorithm with 128-bit or 256-bit encryption.
Step 5	key-string { [<i>0/6/7</i>] <i>pwd-string</i> <i>pwd-string</i>}	Sets the password for a key string. Only hex characters must be entered.
Step 6	lifetime local [<i>start timestamp {hh:mm:ss / day / month / year}</i>] [<i>duration seconds</i> <i>end timestamp {hh:mm:ss / day / month / year}</i>]	Sets the lifetime of the pre shared key.
Step 7	end	Returns to privileged EXEC mode.

Example

Following is an indicative example:

```
Switch(config)# Key chain keychain1 macsec
Switch(config-key-chain)# key 1000
Switch(config-keychain-key)# cryptographic-algorithm gcm-aes-128
Switch(config-keychain-key)# key-string 12345678901234567890123456789012
Switch(config-keychain-key)# lifetime local 12:12:00 July 28 2016 12:19:00 July
28 2016
Switch(config-keychain-key)# end
```

Configuring MACsec MKA on an Interface using PSK



Note To avoid traffic drop across sessions, the **mka policy** command must be configured before the **mka pre-shared-key key-chain** command.

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enters global configuration mode.
Step 2	<code>interface <i>interface-id</i></code>	Enters interface configuration mode.
Step 3	<code>macsec network-link</code>	Enables MACsec on the interface. Note The <code>macsec network-link</code> command does not block MKA sessions for downlink ports. Use the <code>macsec</code> command instead.
Step 4	<code>mka policy <i>policy-name</i></code>	Configures an MKA policy.
Step 5	<code>mka pre-shared-key key-chain <i>key-chain name</i></code>	Configures an MKA pre-shared-key key-chain name. Note The MKA pre-shared key can be configured on either physical interface or sub-interfaces and not on both.
Step 6	<code>macsec replay-protection window-size <i>frame number</i></code>	Sets the MACsec window size for replay protection.
Step 7	<code>end</code>	Returns to privileged EXEC mode.

Example

Following is an indicative example:

```
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# mka policy mka_policy
Switch(config-if)# mka pre-shared-key key-chain key-chain-name
Switch(config-if)# macsec replay-protection window-size 10
Switch(config-if)# end
```

What to do next

It is not recommended to change the MKA policy on an interface with MKA PSK configured when the session is running. However, if a change is required, you must reconfigure the policy as follows:

1. Disable the existing session by removing `macsec network-link` configuration on each of the participating node using the `no macsec network-link` command
2. Configure the MKA policy on the interface on each of the participating node using the `mka policy policy-name` command.
3. Enable the new session on each of the participating node by using the `macsec network-link` command.

Configuring MACsec MKA using EAP-TLS

To configure MACsec with MKA on point-to-point links, perform these tasks:

- Configure Certificate Enrollment
 - Generate Key Pairs
 - Configure SCEP Enrollment
 - Configure Certificates Manually
- Configure an Authentication Policy
- Configure EAP-TLS Profiles and IEEE 802.1x Credentials
- Configure MKA MACsec using EAP-TLS on Interfaces

Remote Authentication

Generating Key Pairs

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	crypto key generate rsa label <i>label-name</i> general-keys modulus <i>size</i>	Generates a RSA key pair for signing and encryption. You can also assign a label to each key pair using the label keyword. The label is referenced by the trustpoint that uses the key pair. If you do not assign a label, the key pair is automatically labeled <Default-RSA-Key>. If you do not use additional keywords this command generates one general purpose RSA key pair. If the modulus is not specified, the default key modulus of 1024 is used. You can specify other modulus sizes with the modulus keyword.
Step 3	end	Returns to privileged EXEC mode.
Step 4	show authentication session interface <i>interface-id</i>	Verifies the authorized session security status.
Step 5	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Enrollment using SCEP

Simple Certificate Enrollment Protocol (SCEP) is a Cisco-developed enrollment protocol that uses HTTP to communicate with the certificate authority (CA) or registration authority (RA). SCEP is the most commonly used method for sending and receiving requests and certificates.

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 3	enrollment url <i>url name pem</i>	Specifies the URL of the CA on which your device should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80. The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 4	rsakeypair <i>label</i>	Specifies which key pair to associate with the certificate. Note The rsakeypair name must match the trust-point name.
Step 5	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 6	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 7	revocation-check crl	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 8	auto-enroll <i>percent regenerate</i>	Enables auto-enrollment, allowing the client to automatically request a rollover certificate from the CA. If auto-enrollment is not enabled, the client must be manually re-enrolled in your PKI upon certificate expiration. By default, only the Domain Name System (DNS) name of the device is included in the certificate. Use the percent argument to specify that a new certificate will be requested after the

	Command or Action	Purpose
		<p>percentage of the lifetime of the current certificate is reached.</p> <p>Use the regenerate keyword to generate a new key for the certificate even if a named key already exists.</p> <p>If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: “! RSA key pair associated with trustpoint is exportable.”</p> <p>It is recommended that a new key pair be generated for security reasons.</p>
Step 9	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 10	exit	Exits global configuration mode.
Step 11	show crypto pki certificate <i>trustpoint name</i>	Displays information about the certificate for the trust point.

Configuring Enrollment Manually

If your CA does not support SCEP or if a network connection between the router and CA is not possible. Perform the following task to set up manual certificate enrollment:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 3	enrollment url <i>url name pem</i>	<p>Specifies the URL of the CA on which your device should send certificate requests.</p> <p>An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80.</p> <p>The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.</p>
Step 4	rsakeypair <i>label</i>	Specifies which key pair to associate with the certificate.

	Command or Action	Purpose
Step 5	<code>serial-number none</code>	The none keyword specifies that a serial number will not be included in the certificate request.
Step 6	<code>ip-address none</code>	The none keyword specifies that no IP address should be included in the certificate request.
Step 7	<code>revocation-check crl</code>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 8	<code>exit</code>	Exits Global Configuration mode.
Step 9	<code>crypto pki authenticate <i>name</i></code>	Retrieves the CA certificate and authenticates it.
Step 10	<code>crypto pki enroll <i>name</i></code>	<p>Generates certificate request and displays the request for copying and pasting into the certificate server.</p> <p>Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request.</p> <p>You are also given the choice about displaying the certificate request to the console terminal.</p> <p>The base-64 encoded certificate with or without PEM headers as requested is displayed.</p>
Step 11	<code>crypto pki import <i>name</i> certificate</code>	<p>Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate.</p> <p>The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from “.req” to “.cert”. For usage key certificates, the extensions “-sign.cert” and “-encr.cert” are used.</p> <p>The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch.</p> <p>Note Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.</p>

	Command or Action	Purpose
Step 12	<code>exit</code>	Exits global configuration mode.
Step 13	<code>show crypto pki certificate <i>trustpoint name</i></code>	Displays information about the certificate for the trust point.
Step 14	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling 802.1x Authentication and Configuring AAA

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>aaa new-model</code>	Enables AAA.
Step 4	<code>dot1x system-auth-control</code>	Enables 802.1X on your device.
Step 5	<code>radius server <i>name</i></code>	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
Step 6	<code>address <i>ip-address</i> auth-port <i>port-number</i></code> <code>acct-port <i>port-number</i></code>	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 7	<code>automate-tester username <i>username</i></code>	Enables the automated testing feature for the RADIUS server. With this practice, the device sends periodic test authentication messages to the RADIUS server. It looks for a RADIUS response from the server. A success message is not necessary - a failed authentication suffices, because it shows that the server is alive.
Step 8	<code>key <i>string</i></code>	Configures the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 9	<code>radius-server deadtime <i>minutes</i></code>	Improves RADIUS response time when some servers might be unavailable and skips unavailable servers immediately.
Step 10	<code>exit</code>	Returns to global configuration mode.

	Command or Action	Purpose
Step 11	aaa group server radius <i>group-name</i>	Groups different RADIUS server hosts into distinct lists and distinct methods, and enters server group configuration mode.
Step 12	server <i>name</i>	Assigns the RADIUS server name.
Step 13	exit	Returns to global configuration mode.
Step 14	aaa authentication dot1x default group <i>group-name</i>	Sets the default authentication server group for IEEE 802.1x.
Step 15	aaa authorization network default group <i>group-name</i>	Sets the network authorization default group.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	eap profile <i>profile-name</i>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	method tls	Enables EAP-TLS method on the device.
Step 5	pki-trustpoint <i>name</i>	Sets the default PKI trustpoint.
Step 6	exit	Returns to global configuration mode.
Step 7	dot1x credentials <i>profile-name</i>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	username <i>username</i>	Sets the authentication user ID.
Step 9	pki-trustpoint <i>name</i>	Sets the default PKI trustpoint.
Step 10	end	Returns to privileged EXEC mode.

Applying the 802.1x MACsec MKA Configuration on Interfaces

To apply MACsec MKA using EAP-TLS to interfaces, perform the following task:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<code>interface <i>interface-id</i></code>	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 3	<code>macsec network-link</code>	Enables MACsec on the interface.
Step 4	<code>authentication periodic</code>	Enables reauthentication for this port.
Step 5	<code>authentication timer reauthenticate interval</code>	Sets the reauthentication interval.
Step 6	<code>access-session host-mode multi-domain</code>	Allows hosts to gain access to the interface.
Step 7	<code>access-session closed</code>	Prevents preauthentication access on the interface.
Step 8	<code>access-session port-control auto</code>	Sets the authorization state of a port.
Step 9	<code>dot1x pae both</code>	Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator.
Step 10	<code>dot1x credentials profile</code>	Assigns a 802.1x credentials profile to the interface.
Step 11	<code>dot1x supplicant eap profile <i>name</i></code>	Assigns the EAP-TLS profile to the interface.
Step 12	<code>service-policy type control subscriber <i>control-policy name</i></code>	Applies a subscriber control policy to the interface.
Step 13	<code>exit</code>	Returns to privileged EXEC mode.
Step 14	<code>show macsec interface</code>	Displays MACsec details for the interface.
Step 15	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Local Authentication

Configuring the EAP Credentials using Local Authentication

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>aaa new-model</code>	Enables AAA.

	Command or Action	Purpose
Step 4	aaa local authentication default authorization default	Sets the default local authentication and default local authorization method.
Step 5	aaa authentication dot1x default local	Sets the default local username authentication list for IEEE 802.1x.
Step 6	aaa authorization network default local	Sets an authorization method list for local user.
Step 7	aaa authorization credential-download default local	Sets an authorization method list for use of local credentials.
Step 8	exit	Returns to privileged EXEC mode.

Configuring the Local EAP-TLS Authentication and Authorization Profile

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	aaa new-model	Enables AAA.
Step 4	dot1x credentials <i>profile-name</i>	Configures the dot1x credentials profile and enters dot1x credentials configuration mode.
Step 5	username <i>name</i> password <i>password</i>	Sets the authentication user ID and password.
Step 6	exit	Returns to global configuration mode.
Step 7	aaa attribute list <i>list-name</i>	(Optional) Sets the AAA attribute list definition and enters attribute list configuration mode.
Step 8	aaa attribute type linksec-policy must-secure	(Optional) Specifies the AAA attribute type.
Step 9	exit	Returns to global configuration mode.
Step 10	username <i>name</i> aaa attribute list <i>name</i>	(Optional) Specifies the AAA attribute list for the user ID.
Step 11	end	Returns to privileged EXEC mode.

Configuring Enrollment using SCEP

Simple Certificate Enrollment Protocol (SCEP) is a Cisco-developed enrollment protocol that uses HTTP to communicate with the certificate authority (CA) or registration authority (RA). SCEP is the most commonly used method for sending and receiving requests and certificates.

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	enrollment url <i>url name pem</i>	Specifies the URL of the CA on which your device should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80. The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.
Step 5	rsakeypair <i>label</i>	Specifies which key pair to associate with the certificate. Note The rsakeypair name must match the trust-point name.
Step 6	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	revocation-check <i>crl</i>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	auto-enroll <i>percent regenerate</i>	Enables auto-enrollment, allowing the client to automatically request a rollover certificate from the CA. If auto-enrollment is not enabled, the client must be manually re-enrolled in your PKI upon certificate expiration. By default, only the Domain Name System (DNS) name of the device is included in the certificate.

	Command or Action	Purpose
		<p>Use the percent argument to specify that a new certificate will be requested after the percentage of the lifetime of the current certificate is reached.</p> <p>Use the regenerate keyword to generate a new key for the certificate even if a named key already exists.</p> <p>If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: “! RSA key pair associated with trustpoint is exportable.”</p> <p>It is recommended that a new key pair be generated for security reasons.</p>
Step 10	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 11	exit	Exits global configuration mode.
Step 12	show crypto pki certificate <i>trustpoint name</i>	Displays information about the certificate for the trust point.

Configuring Enrollment Manually

If your CA does not support SCEP or if a network connection between the router and CA is not possible. Perform the following task to set up manual certificate enrollment:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	enrollment url <i>url name pem</i>	<p>Specifies the URL of the CA on which your device should send certificate requests.</p> <p>An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80.</p> <p>The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.</p>

	Command or Action	Purpose
Step 5	<code>rsa keypair <i>label</i></code>	Specifies which key pair to associate with the certificate.
Step 6	<code>serial-number none</code>	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	<code>ip-address none</code>	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	<code>revocation-check <i>crl</i></code>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	<code>exit</code>	Exits Global Configuration mode.
Step 10	<code>crypto pki authenticate <i>name</i></code>	Retrieves the CA certificate and authenticates it.
Step 11	<code>crypto pki enroll <i>name</i></code>	<p>Generates certificate request and displays the request for copying and pasting into the certificate server.</p> <p>Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request.</p> <p>You are also given the choice about displaying the certificate request to the console terminal.</p> <p>The base-64 encoded certificate with or without PEM headers as requested is displayed.</p>
Step 12	<code>crypto pki import <i>name certificate</i></code>	<p>Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate.</p> <p>The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from “.req” to “.crt”. For usage key certificates, the extensions “-sign.crt” and “-encr.crt” are used.</p> <p>The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch.</p>

	Command or Action	Purpose
		Note Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.
Step 13	<code>exit</code>	Exits Global Configuration mode.
Step 14	<code>show crypto pki certificate trustpoint name</code>	Displays information about the certificate for the trust point.
Step 15	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>eap profile profile-name</code>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	<code>method tls</code>	Enables EAP-TLS method on the device.
Step 5	<code>pki-trustpoint name</code>	Sets the default PKI trustpoint.
Step 6	<code>exit</code>	Returns to global configuration mode.
Step 7	<code>dot1x credentials profile-name</code>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	<code>username username</code>	Sets the authentication user ID.
Step 9	<code>pki-trustpoint name</code>	Sets the default PKI trustpoint.
Step 10	<code>end</code>	Returns to privileged EXEC mode.

Applying the 802.1x MKA MACsec Configuration on Interfaces

To apply MKA MACsec using EAP-TLS to interfaces, perform the following task:

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>interface <i>interface-id</i></code>	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	<code>macsec</code>	Enables MACsec on the interface.
Step 5	<code>authentication periodic</code>	Enables reauthentication for this port.
Step 6	<code>authentication timer reauthenticate interval</code>	Sets the reauthentication interval.
Step 7	<code>access-session host-mode multi-domain</code>	Allows hosts to gain access to the interface.
Step 8	<code>access-session closed</code>	Prevents preauthentication access on the interface.
Step 9	<code>access-session port-control auto</code>	Sets the authorization state of a port.
Step 10	<code>dot1x pae both</code>	Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator.
Step 11	<code>dot1x credentials profile</code>	Assigns a 802.1x credentials profile to the interface.
Step 12	<code>dot1x authenticator eap profile <i>name</i></code>	Assigns the EAP-TLS authenticator profile to the interface.
Step 13	<code>dot1x supplicant eap profile <i>name</i></code>	Assigns the EAP-TLS supplicant profile to the interface.
Step 14	<code>service-policy type control subscriber <i>control-policy name</i></code>	Applies a subscriber control policy to the interface.
Step 15	<code>exit</code>	Returns to privileged EXEC mode.
Step 16	<code>show macsec interface</code>	Displays MACsec details for the interface.
Step 17	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Verifying MACsec MKA using EAP-TLS

Use the following **show** commands to verify the configuration of MACsec MKA using EAP-TLS. Given below are the sample outputs of the **show** commands.


```

                Status: Authorized
                Domain: DATA
    Oper host mode: multi-host
    Oper control dir: both
    Session timeout: N/A
    Common Session ID: 00000000000000BB72E8AFA
    Acct Session ID: Unknown
                Handle: 0xc3000001
    Current Policy: MUSTS_1

```

```

Local Policies:
  Security Policy: Must Secure
  Security Status: Link Secured

```

```
Server Policies:
```

```

Method status list:
  Method          State
  dot1xSup        Authc Success
  dot1x            Authc Success

```

Configuring Cisco TrustSec MACsec

Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode

Before you begin

When manually configuring Cisco TrustSec on an interface, consider these usage guidelines and restrictions:

- If no SAP parameters are defined, Cisco TrustSec encapsulation or encryption is not performed.
- If you select GCM as the SAP operating mode, you must have a MACsec Encryption software license from Cisco. If you select GCM without the required license, the interface is forced to a link-down state.
- These protection levels are supported when you configure SAP pairwise master key (`sap pmk`):
 - SAP is not configured—no protection.
 - **`sap mode-list gcm-encrypt gmac no-encap`**—protection desirable but not mandatory.
 - **`sap mode-list gcm-encrypt gmac`**—confidentiality preferred and integrity required. The protection is selected by the supplicant according to supplicant preference.
 - **`sap mode-list gmac`**—integrity only.
 - **`sap mode-list gcm-encrypt`**—confidentiality required.
 - **`sap mode-list gmac gcm-encrypt`**—integrity required and preferred, confidentiality optional.
- When CTS is configured on an interface and the System MTU is set to a value greater than 9191, the resulting packet size is limited to 9190.
- Before changing the configuration from MKA to Cisco TrustSec SAP and vice versa, we recommend that you remove the interface configuration.

Beginning in privileged EXEC mode, follow these steps to manually configure Cisco TrustSec on an interface to another Cisco TrustSec device:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface tengigabitethernet 1/1/2	Note Enters interface configuration mode.
Step 3	cts manual Example: Switch(config-if)# cts manual	Enters Cisco TrustSec manual configuration mode.
Step 4	sap pmk key [mode-list mode1 [mode2 [mode3 [mode4]]]] Example: Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap	<p>(Optional) Configures the SAP pairwise master key (PMK) and operation mode. SAP is disabled by default in Cisco TrustSec manual mode.</p> <ul style="list-style-type: none"> • <i>key</i>—A hexadecimal value with an even number of characters and a maximum length of 32 characters. <p>The SAP operation mode options:</p> <ul style="list-style-type: none"> • gcm-encrypt—Authentication and encryption <p>Note Select this mode for MACsec authentication and encryption if your software license supports MACsec encryption.</p> <ul style="list-style-type: none"> • gmac—Authentication, no encryption • no-encap—No encapsulation • null—Encapsulation, no authentication or encryption <p>Note If the interface is not capable of data link encryption, no-encap is the default and the only available SAP operating mode. SGT is not supported.</p>

	Command or Action	Purpose
Step 5	no propagate sgt Example: Switch(config-if-cts-manual)# no propagate sgt	Use the no form of this command when the peer is incapable of processing a SGT. The no propagate sgt command prevents the interface from transmitting the SGT to the peer.
Step 6	exit Example: Switch(config-if-cts-manual)# exit	Exits Cisco TrustSec 802.1x interface configuration mode.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 8	show cts interface [<i>interface-id</i> brief summary]	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.

Example

This example shows how to configure Cisco TrustSec authentication in manual mode on an interface:

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# end
```

Configuration Examples for MACsec Encryption

Configuring Switch-to-host MACsec Encryption

Follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter the password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch> configure terminal	Enters the global configuration mode.
Step 3	interface interface-id	Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	switchport access vlan vlan-id	Configure the access VLAN for the port.
Step 5	switchport mode access	Configure the interface as an access port.
Step 6	macsec	Enable 802.1ae MACsec on the interface. The macsec command enables MKA MACsec on switch-to-host links (downlink ports) only.
Step 7	authentication event linksec fail action authorize vlan vlan-id	(Optional) Specify that the switch processes authentication link-security failures resulting from unrecognized user credentials by authorizing a restricted VLAN on the port after a failed authentication attempt.
Step 8	authentication host-mode multi-domain	Configure authentication manager mode on the port to allow both a host and a voice device to be authenticated on the 802.1x-authorized port. If not configured, the default host mode is single.
Step 9	authentication linksec policy must-secure	Set the LinkSec security policy to secure the session with MACsec if the peer is available. If not set, the default is <i>should secure</i> .
Step 10	authentication port-control auto	Enable 802.1x authentication on the port. The port changes to the authorized or unauthorized state based on the authentication exchange between the switch and the client.
Step 11	authentication periodic	Enable or Disable Reauthentication for this port .
Step 12	authentication timer reauthenticate	Enter a value between 1 and 65535 (in seconds). Obtains re-authentication timeout value from the server. Default re-authentication time is 3600 seconds.
Step 13	authentication violation protect	Configure the port to drop unexpected incoming MAC addresses when a new device connects to a port or when a device connects to a port after the maximum number of devices are connected to that port. If not configured, the default is to shut down the port.

	Command or Action	Purpose
Step 14	mka policy <i>policy name</i>	Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering the mka policy global configuration command).
Step 15	dot1x pae authenticator	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 16	spanning-tree portfast	Enable spanning tree Port Fast on the interface in all its associated VLANs. When Port Fast feature is enabled, the interface changes directly from a blocking state to a forwarding state without making the intermediate spanning-tree state changes
Step 17	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 18	show authentication session interface <i>interface-id</i>	Verify the authorized session security status.
Step 19	show authentication session interface <i>interface-id</i> details	Verify the details of the security status of the authorized session.
Step 20	show macsec interface <i>interface-id</i>	Verify MacSec status on the interface.
Step 21	show mka sessions	Verify the established mka sessions.
Step 22	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Examples for MACsec MKA using EAP-TLS

Example: Enrolling the Certificate

Configure Crypto PKI Trustpoint:

```
crypto pki trustpoint POLESTAR-IOS-CA
  enrollment terminal
  subject-name CN=ASR1000x1@polestar.com, C=IN, ST=KA, OU=ENG,O=Polestar
  revocation-check none
  rsakeypair mkaioscarsa
  storage nvram:
!
```

Manual Installation of Root CA certificate:

```
crypto pki authenticate POLESTAR-IOS-CA
```

Example: Enabling 802.1x Authentication and AAA Configuration

```

aaa new-model
dot1x system-auth-control
radius server ISE
  address ipv4 <ISE ipv4 address> auth-port 1645 acct-port 1646
  automate-tester username dummy
  key dummy123
  radius-server deadtime 2
!
aaa group server radius ISEGRP
  server name ISE
!
aaa authentication dot1x default group ISEGRP
aaa authorization network default group ISEGRP

```

Example: Configuring EAP-TLS Profile and 802.1X Credentials

```

eap profile EAPTLS-PROF-IOSCA
  method tls
  pki-trustpoint POLESTAR-IOS-CA
!

dot1x credentials EAPTLSCRED-IOSCA
  username asr1000@polestar.company.com
  pki-trustpoint POLESTAR-IOS-CA
!

```

Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface

```

interface TenGigabitEthernet0/1
  macsec network-link
  authentication periodic
  authentication timer reauthenticate <reauthentication interval>
  access-session host-mode multi-host
  access-session closed
  access-session port-control auto
  dot1x pae both
  dot1x credentials EAPTLSCRED-IOSCA
  dot1x supplicant eap profile EAPTLS-PROF-IOSCA
  service-policy type control subscriber DOT1X_POLICY_RADIUS

```

Example: Displaying MKA Information

The following is sample output from the **show mka sessions** command.

```

Device# show mka sessions

Total MKA Sessions..... 1
  Secured Sessions... 1
  Pending Sessions... 0

```

```

=====
Interface         Local-TxSCI             Policy-Name             Inherited
Key-Server
Port-ID           Peer-RxSCI              MACsec-Peers            Status                   CKN
=====
Gil1/0/1         204c.9e85.ede4/002b p2                NO                       Secured                   YES
43                c800.8459.e764/002a 1
010000000000000000000000000000000000000000000000000000000000000000000000000000000000

```

The following is sample output from the **show mka sessions interface interface-name** command.

Device# **show mka sessions interface GigabitEthernet 1/0/1**

Summary of All Currently Active MKA Sessions on Interface
GigabitEthernet1/0/1...

```

=====
Interface         Local-TxSCI             Policy-Name             Inherited
Key-Server
Port-ID           Peer-RxSCI              MACsec-Peers            Status                   CKN
=====
Gil1/0/1         204c.9e85.ede4/002b p2                NO                       Secured                   YES
43                c800.8459.e764/002a 1
010000000000000000000000000000000000000000000000000000000000000000000000000000000000

```

The following is sample output from the **show mka sessions interface interface-name detail** command.

Device# **show mka sessions interface GigabitEthernet 1/0/1 detail**

MKA Detailed Status for MKA Session

=====

Status: SECURED - Secured MKA Session with MACsec

```

Local Tx-SCI..... 204c.9e85.ede4/002b
Interface MAC Address.... 204c.9e85.ede4
MKA Port Identifier..... 43
Interface Name..... GigabitEthernet1/0/1
Audit Session ID.....
CAK Name (CKN).....
010000000000000000000000000000000000000000000000000000000000000000000000000000000000
Member Identifier (MI)... D46CBEC05D5D67594543CEAE
Message Number (MN)..... 89567
EAP Role..... NA
Key Server..... YES
MKA Cipher Suite..... AES-128-CMAC

Latest SAK Status..... Rx & Tx
Latest SAK AN..... 0
Latest SAK KI (KN)..... D46CBEC05D5D67594543CEAE00000001 (1)
Old SAK Status..... FIRST-SAK
Old SAK AN..... 0
Old SAK KI (KN)..... FIRST-SAK (0)

```



```

Latest SAK AN..... 0
Latest SAK KI (KN)..... D46CBEC05D5D67594543CEAE00000001 (1)
Old SAK Status..... FIRST-SAK
Old SAK AN..... 0
Old SAK KI (KN)..... FIRST-SAK (0)

SAK Transmit Wait Time... 0s (Not waiting for any peers to respond)
SAK Retire Time..... 0s (No Old SAK to retire)

MKA Policy Name..... p2
Key Server Priority..... 2
Delay Protection..... NO
Replay Protection..... YES
Replay Window Size..... 0
Confidentiality Offset... 0
Algorithm Agility..... 80C201
Send Secure Announcement.. DISABLED
SAK Cipher Suite..... 0080C20001000001 (GCM-AES-128)
MACsec Capability..... 3 (MACsec Integrity, Confidentiality, & Offset)
MACsec Desired..... YES

# of MACsec Capable Live Peers..... 1
# of MACsec Capable Live Peers Responded.. 1
    
```

```

Live Peers List:
  MI                MN                Rx-SCI (Peer)          KS Priority
  -----
  38046BA37D7DA77E06D006A9  89560          c800.8459.e764/002a   10
    
```

```

Potential Peers List:
  MI                MN                Rx-SCI (Peer)          KS Priority
  -----
    
```

```

Dormant Peers List:
  MI                MN                Rx-SCI (Peer)          KS Priority
  -----
    
```

The following is sample output from the **show mka policy** command:

```
Device# show mka policy
```

MKA Policy Summary...

Policy Interfaces Name Applied	KS Priority	Delay Protect	Replay Protect	Window Size	Conf Offset	Cipher Suite(s)
DEFAULT POLICY	0	FALSE	TRUE	0	0	GCM-AES-128
p1	1	FALSE	TRUE	0	0	GCM-AES-128

```
p2                2                FALSE    TRUE    0        0        GCM-AES-128
  Gil/0/1
```

The following is sample output from the **show mka policy *policy-name*** command:

```
Device# show mka policy p2
```

```
MKA Policy Summary...
```

Policy	KS	Delay	Replay	Window	Conf	Cipher
Interfaces						
Name	Priority	Protect	Protect	Size	Offset	Suite(s)
Applied						
p2	2	FALSE	TRUE	0	0	GCM-AES-128
Gil/0/1						

The following is sample output from the **show mka policy *policy-name* detail** command:

```
Device# show mka policy p2 detail
```

```
MKA Policy Configuration ("p2")
=====
MKA Policy Name..... p2
Key Server Priority.... 2
Confidentiality Offset. 0
Send Secure Announcement..DISABLED
Cipher Suite(s)..... GCM-AES-128
```

```
Applied Interfaces...
  GigabitEthernet1/0/1
```

The following is sample output from the **show mka statistics interface *interface-name*** command:

```
Device# show mka statistics interface GigabitEthernet 1/0/1
```

```
MKA Statistics for Session
=====
Reauthentication Attempts.. 0
```

```
CA Statistics
  Pairwise CAKs Derived... 0
  Pairwise CAK Rekeys..... 0
  Group CAKs Generated.... 0
  Group CAKs Received..... 0
```

```
SA Statistics
  SAKs Generated..... 1
  SAKs Rekeyed..... 0
  SAKs Received..... 0
  SAK Responses Received.. 1
```

```
MKPDU Statistics
```



```

"Distributed SAK"..... 1
"Distributed CAK"..... 0

MKA Error Counter Totals
=====
Session Failures
  Bring-up Failures..... 0
  Reauthentication Failures..... 0
  Duplicate Auth-Mgr Handle..... 0

SAK Failures
  SAK Generation..... 0
  Hash Key Generation..... 0
  SAK Encryption/Wrap..... 0
  SAK Decryption/Unwrap..... 0
  SAK Cipher Mismatch..... 0

CA Failures
  Group CAK Generation..... 0
  Group CAK Encryption/Wrap..... 0
  Group CAK Decryption/Unwrap..... 0
  Pairwise CAK Derivation..... 0
  CKN Derivation..... 0
  ICK Derivation..... 0
  KEK Derivation..... 0
  Invalid Peer MACsec Capability... 0

MACsec Failures
  Rx SC Creation..... 0
  Tx SC Creation..... 0
  Rx SA Installation..... 0
  Tx SA Installation..... 0

MKPDU Failures
  MKPDU Tx..... 0
  MKPDU Rx Validation..... 0
  MKPDU Rx Bad Peer MN..... 0
  MKPDU Rx Non-recent Peerlist MN.. 0

```

Cisco TrustSec Switch-to-Switch Link Security Configuration Example

This example shows the configuration necessary for a seed and non-seed device for Cisco TrustSec switch-to-switch security. You must configure the AAA and RADIUS for link security. In this example, ACS-1 through ACS-3 can be any server names and cts-radius is the Cisco TrustSec server.

Seed Device Configuration:

```

Switch(config)#aaa new-model
Switch(config)#radius server ACS-1
Switch(config-radius-server)#address ipv4 10.5.120.12 auth-port 1812 acct-port 1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit

```

```

Switch(config)#radius server ACS-2
Switch(config-radius-server)#address ipv4 10.5.120.14 auth-port 1812 acct-port
1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit
Switch(config)#radius server ACS-3
Switch(config-radius-server)#address ipv4 10.5.120.15 auth-port 1812 acct-port
1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit
Switch(config)#aaa group server radius cts-radius
Switch(config-sg-radius)#server name ACS-1
Switch(config-sg-radius)#server name ACS-2
Switch(config-sg-radius)#server name ACS-3
Switch(config-sg-radius)#exit
Switch(config)#aaa authentication login default none
Switch(config)#aaa authentication dot1x default group cts-radius
Switch(config)#aaa authorization network cts-radius group cts-radius
Switch(config)#aaa session-id common
Switch(config)#cts authorization list cts-radius
Switch(config)#dot1x system-auth-control

Switch(config)#interface gil/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac

Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)#interface gil/1/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#cts manual
Switch(config-if-cts-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)#radius-server vsa send authentication
Switch(config)#end
Switch#cts credentials id cts-36 password trustsec123

```

Non-Seed Device:

```

Switch(config)#aaa new-model
Switch(config)#aaa session-id common
Switch(config)#dot1x system-auth-control

Switch(config)#interface gil/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac

```

```
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)#interface gil/1/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts manual
Switch(config-if-cts-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

Switch(config)#radius-server vsa send authentication
Switch(config)#end
Switch(config)#cts credentials id cts-72 password trustsec123
```



CHAPTER 7

Configuring Local Authentication and Authorization

- [Finding Feature Information, on page 137](#)
- [How to Configure Local Authentication and Authorization, on page 137](#)
- [Monitoring Local Authentication and Authorization, on page 139](#)
- [Additional References, on page 140](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

How to Configure Local Authentication and Authorization

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.



Note

To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

Follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa new-model Example: <pre>Device (config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authentication login default local Example: <pre>Device (config)# aaa authentication login default local</pre>	Sets the login authentication to use the local username database. The default keyword applies the local user database authentication to all ports.
Step 5	aaa authorization exec local Example: <pre>Device (config)# aaa authorization exec local</pre>	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 6	aaa authorization network local Example: <pre>Device (config)# aaa authorization network local</pre>	Configures user AAA authorization for all network-related service requests.
Step 7	username name [privilege level] {password encryption-type password} Example: <pre>Device (config)# username your_user_name privilege 1 password 7 secret567</pre>	Enters the local database, and establishes a username-based authentication system. <p>Repeat this command for each user.</p> <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word. Spaces and quotation marks are not allowed.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For <i>password</i>, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 8

Configuring Secure Shell (SSH)

- [Finding Feature Information, on page 141](#)
- [Prerequisites for Configuring Secure Shell, on page 141](#)
- [Restrictions for Configuring Secure Shell, on page 142](#)
- [Information About Configuring Secure Shell, on page 142](#)
- [How to Configure Secure Shell, on page 144](#)
- [Monitoring the SSH Configuration and Status, on page 148](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for Configuring Secure Shell

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an Rivest, Shamir, and Adleman (RSA) public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.
- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.
- SCP relies on SSH for security.
- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.

- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.
- The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.)
- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for Configuring Secure Shell

The following are restrictions for configuring the Device for secure shell.

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- The Device supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.
- The **-l** keyword and **userid** : {number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.

Information About Configuring Secure Shell

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 2 (SSHv2).

SSH And Switch Access

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 2 (SSHv2).

SSH functions the same in IPv6 as in IPv4. For IPv6, SSH supports IPv6 addresses and enables secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

SSH Servers, Integrated Clients, and Supported Versions

The Secure Shell (SSH) Integrated Client feature is an application that runs over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running the SSH server. This connection provides functionality similar to that of an outbound Telnet connection except that the connection is encrypted. With authentication and encryption, the SSH client allows for secure communication over an unsecured network.

The SSH server and SSH integrated client are applications that run on the switch. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client works with publicly and commercially available SSH servers. The SSH client supports the ciphers of Data Encryption Standard (DES), 3DES, and password authentication.



Note The SSH client functionality is available only when the SSH server is enabled.

User authentication is performed like that in the Telnet session to the device. SSH also supports the following user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

SSH Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If the SSH server is running on an active switch and the active switch fails, the new active switch uses the RSA key pair generated by the previous active switch.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command.
- When generating the RSA key pair, the message No host name specified might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Secure Copy Protocol Overview

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.



Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure Copy Protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying device configurations or switch image files. The behavior of SCP is similar to that of remote copy (r`cp`), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the device can determine whether the user has the correct privilege level. To configure the Secure Copy feature, you should understand the SCP concepts.

How to Configure Secure Shell

Setting Up the Device to Run SSH

Follow these steps to set up your Device to run SSH:

Before you begin

Configure user authentication for local or remote access. This step is required. For more information, see Related Topics below.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p><code>hostname hostname</code></p> <p>Example:</p> <p>Device(config)# <code>hostname your_hostname</code></p>	<p>Configures a hostname and IP domain name for your Device.</p> <p>Note Follow this procedure only if you are configuring the Device as an SSH server.</p>
Step 4	<p><code>ip domain-name domain_name</code></p> <p>Example:</p> <p>Device(config)# <code>ip domain-name your_domain</code></p>	Configures a host domain for your Device.
Step 5	<p><code>crypto key generate rsa</code></p> <p>Example:</p> <p>Device(config)# <code>crypto key generate rsa</code></p>	<p>Enables the SSH server for local and remote authentication on the Device and generates an RSA key pair. Generating an RSA key pair for the Device automatically enables SSH.</p> <p>We recommend that a minimum modulus size of 1024 bits.</p> <p>When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.</p> <p>Note Follow this procedure only if you are configuring the Device as an SSH server.</p>
Step 6	<p><code>end</code></p> <p>Example:</p> <p>Device(config)# <code>end</code></p>	Returns to privileged EXEC mode.
Step 7	<p><code>show running-config</code></p> <p>Example:</p> <p>Device# <code>show running-config</code></p>	Verifies your entries.
Step 8	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <p>Device# <code>copy running-config</code></p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	<code>startup-config</code>	

Configuring the SSH Server

Follow these steps to configure the SSH server:



Note This procedure is only required if you are configuring the Device as an SSH server.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip ssh version [2] Example: Device(config)# <code>ip ssh version 2</code>	(Optional) Configures the Device to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client.
Step 4	ip ssh {time-out <i>seconds</i> authentication-retries <i>number</i>} Example: Device(config)# <code>ip ssh time-out 90</code> OR Device(config)# <code>ip ssh authentication-retries 2</code>	Configures the SSH control parameters: <ul style="list-style-type: none"> • time-out <i>seconds</i>: Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the Device uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • authentication-retries <i>number</i>: Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. <p>Repeat this step when configuring both parameters.</p>
Step 5	<p>Use one or both of the following:</p> <ul style="list-style-type: none"> • <code>line vty</code> <code>line_number[ending_line_number]</code> • transport input ssh <p>Example:</p> <pre>Device(config) # line vty 1 10</pre> <p>or</p> <pre>Device(config-line) # transport input ssh</pre>	<p>(Optional) Configures the virtual terminal line settings.</p> <ul style="list-style-type: none"> • Enters line configuration mode to configure the virtual terminal line settings. For the <i>line_number</i> and <i>ending_line_number</i> arguments, the range is from 0 to 15. • Specifies that the Device prevents non-SSH Telnet connections, limiting the device to only SSH connections.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config-line) # end</pre>	Exits line configuration mode and returns to privileged EXEC mode.
Step 7	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show ip ssh • show ssh <p>Example:</p> <pre>Device# show ip ssh</pre> <p>or</p> <pre>Device# show ssh</pre>	<ul style="list-style-type: none"> • Shows the version and configuration information for your SSH server. • Shows the status of the SSH server connections on the Device.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config</pre>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	<code>startup-config</code>	

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 17: Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
<code>show ip ssh</code>	Shows the version and configuration information for the SSH server.
<code>show ssh</code>	Shows the status of the SSH server.



CHAPTER 9

X.509v3 Certificates for SSH Authentication

- [X.509v3 Certificates for SSH Authentication, on page 149](#)
- [Information About X.509v3 Certificates for SSH Authentication, on page 150](#)
- [How to Configure X.509v3 Certificates for SSH Authentication, on page 150](#)
- [Configuration Examples for X.509v3 Certificates for SSH Authentication, on page 154](#)
- [Additional References for X.509v3 Certificates for SSH Authentication, on page 155](#)
- [Feature Information for X.509v3 Certificates for SSH Authentication, on page 156](#)

X.509v3 Certificates for SSH Authentication

The X.509v3 Certificates for secure shell (SSH) Authentication feature uses the X.509v3 digital certificates in server and user authentication at the SSH server side.

Prerequisites for Digital Certificates for SSH Authentication

The Digital Certificates for SSH Authentication feature introduces the **ip ssh server algorithm authentication** command to replace the **ip ssh server authenticate user** command. If you use the **ip ssh server authenticate user** command, the following deprecation message is displayed.

```
Warning: SSH command accepted but this CLI will be deprecated soon. Please move to new CLI
"ip ssh server algorithm authentication". Please configure "default ip ssh server
authenticate user" to make CLI ineffective.
```

Use the **default ip ssh server authenticate user** command to remove the **ip ssh server authenticate user** command from effect. The IOS secure shell (SSH) server then starts using the **ip ssh server algorithm authentication** command.

Restrictions for X.509v3 Certificates for SSH Authentication

The following restrictions are applicable for X.509v3 Certificate for SSH Authentication:

- The X.509v3 Certificates for SSH Authentication feature implementation is applicable only on the IOS secure shell (SSH) server side.
- IOS SSH server supports only the x509v3-ssh-rsa algorithm based certificate for server and user authentication on the IOS SSH server side.

The X.509v3 Certificate for SSH Authentication fails in the following conditions:

- When root certification authority is configured as a trustpoint on the device.
- When a client passes a certificate chain that leads to a self-signed root certificate authority that includes a client certificate, sub-ca certificate, and self-signed root certificate authority.
- When a sub-ca certification is configured as a trustpoint on the device but not included as a trustpoint on the user certificate.

Information About X.509v3 Certificates for SSH Authentication

The following section provides information about digital certificates, and server and user authentication.

Digital Certificates

The validity of the authentication depends upon the strength of the linkage between the public signing key and the identity of the signer. Digital certificates in the X.509v3 format (RFC5280) are used to provide identity management. A chain of signatures by a trusted root certification authority and its intermediate certificate authorities binds a given public signing key to a given digital identity.

Public key infrastructure (PKI) trustpoint helps manage the digital certificates. The association between the certificate and the trustpoint helps track the certificate. The trustpoint contains information about the certificate authority (CA), different identity parameters, and the digital certificate. Multiple trustpoints can be created to associate with different certificates.

Server and User Authentication using X.509v3

For server authentication, the IOS secure shell (SSH) server sends its own certificate to the SSH client for verification. This server certificate is associated with the trustpoint configured in the server certificate profile (ssh-server-cert-profile-server configuration mode).

For user authentication, the SSH client sends the user's certificate to the IOS SSH server for verification. The SSH server validates the incoming user certificate using public key infrastructure (PKI) trustpoints configured in the server certificate profile (ssh-server-cert-profile-user configuration mode).

By default, certificate-based authentication is enabled for server and user at the IOS SSH server end.

How to Configure X.509v3 Certificates for SSH Authentication

The following section provides information about how to configure X.509v3 Certificates for SSH Authentication.

Configuring IOS SSH Server to Use Digital Certificates for Sever Authentication

The following section provides information about Configuring IOS SSH Server to Use Digital Certificates for Sever Authentication.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} Example: Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa	Defines the order of host key algorithms. Only the configured algorithm is negotiated with the secure shell (SSH) client. Note The IOS SSH server must have at least one configured host key algorithm: <ul style="list-style-type: none"> • ssh-rsa – public key based authentication • x509v3-ssh-rsa – certificate-based authentication
Step 4	ip ssh server certificate profile Example: Device(config)# ip ssh server certificate profile	Configures server certificate profile and user certificate profile and enters SSH certificate profile configuration mode.
Step 5	server Example: Device(ssh-server-cert-profile)# server	Configures server certificate profile and enters SSH server certificate profile server configuration mode.
Step 6	trustpoint sign <i>PKI-trustpoint-name</i> Example: Device(ssh-server-cert-profile-server)# trustpoint sign trust1	Attaches the public key infrastructure (PKI) trustpoint to the server certificate profile. The SSH server uses the certificate associated with this PKI trustpoint for server authentication.
Step 7	ocsp-response include Example: Device(ssh-server-cert-profile-server)# ocsp-response include	(Optional) Sends the Online Certificate Status Protocol (OCSP) response or OCSP stapling along with the server certificate. Note By default the “no” form of this command is configured and no OCSP response is sent along with the server certificate.

	Command or Action	Purpose
Step 8	end Example: <pre>Device (ssh-server-cert-profile-server) # end</pre>	Exits SSH server certificate profile server configuration mode and enters privileged EXEC mode.

Configuring IOS SSH Server to Verify User's Digital Certificate for User Authentication

The following section provides information about configuring IOS SSH Server to Verify User's Digital Certificate for User Authentication.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip ssh server algorithm authentication {publickey keyboard password} Example: <pre>Device(config)# ip ssh server algorithm authentication publickey</pre>	Defines the order of user authentication algorithms. Only the configured algorithm is negotiated with the secure shell (SSH) client. Note <ul style="list-style-type: none"> The IOS SSH server must have at least one configured user authentication algorithm. To use the certificate method for user authentication, the publickey keyword must be configured. The ip ssh server algorithm authentication command replaces the ip ssh server authenticate user command.
Step 4	ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]}	Defines the order of public key algorithms. Only the configured algorithm is accepted by the SSH client for user authentication.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa</pre>	<p>Note The IOS SSH client must have at least one configured public key algorithm:</p> <ul style="list-style-type: none"> • ssh-rsa – public-key-based authentication • x509v3-ssh-rsa – certificate-based authentication
Step 5	<p>ip ssh server certificate profile</p> <p>Example:</p> <pre>Device(config)#ip ssh server certificate profile</pre>	Configures server certificate profile and user certificate profile and enters SSH certificate profile configuration mode.
Step 6	<p>user</p> <p>Example:</p> <pre>Device(ssh-server-cert-profile)# user</pre>	Configures user certificate profile and enters SSH server certificate profile user configuration mode.
Step 7	<p>trustpoint verify <i>PKI-trustpoint-name</i></p> <p>Example:</p> <pre>Device(ssh-server-cert-profile-user)#trustpoint verify trust2</pre>	<p>Configures the public key infrastructure (PKI) trustpoint that is used to verify the incoming user certificate.</p> <p>Note Configure multiple trustpoints by executing the same command multiple times. A maximum of 10 trustpoints can be configured.</p>
Step 8	<p>ocsp-response required</p> <p>Example:</p> <pre>Device(ssh-server-cert-profile-user)# ocsp-response required</pre>	<p>(Optional) Mandates the presence of the Online Certificate Status Protocol (OCSP) response with the incoming user certificate.</p> <p>Note By default the “no” form of this command is configured and the user certificate is accepted without an OCSP response.</p>
Step 9	<p>end</p> <p>Example:</p> <pre>Device(ssh-server-cert-profile-user)#end</pre>	Exits SSH server certificate profile user configuration mode and enters privileged EXEC mode.

Verifying Configuration for Server and User Authentication Using Digital Certificates

The following section provides information about verifying configuration for Server and User Authentication Using Digital Certificates.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip ssh Example: Device# show ip ssh SSH Enabled - version 1.99 Authentication methods:publickey,keyboard-interactive,password Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa Authentication timeout: 120 secs; Authentication retries: 3 Minimum expected Diffie Hellman key size : 1024 bits	Displays the currently configured authentication methods. To confirm the use of certificate-based authentication, ensure that the x509v3-ssh-rsa algorithm is the configured host key algorithm.

Configuration Examples for X.509v3 Certificates for SSH Authentication

The following section provides examples for user and server authentication using digital certificates.

Example: Configuring IOS SSH Server to Use Digital Certificates for Server Authentication

This example shows how to configure IOS SSH Server to Use Digital Certificates for Server Authentication.

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
```



```
Device(ssh-server-cert-profile)# server
Device(ssh-server-cert-profile-server)# trustpoint sign trust1
Device(ssh-server-cert-profile-server)# exit
```

Example: Configuring IOS SSH Server to Verify User's Digital Certificate for User Authentication

This example shows how to configure IOS SSH Server to Verify User's Digital Certificate for User Authentication.

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm authentication publickey
Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(ssh-server-cert-profile)# user
Device(ssh-server-cert-profile-user)# trustpoint verify trust2
Device(ssh-server-cert-profile-user)# end
```

Additional References for X.509v3 Certificates for SSH Authentication

Related Documents

Related Topic	Document Title
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
SSH authentication	“Secure Shell-Configuring User Authentication Methods” chapter in <i>Secure Shell Configuration Guide</i>
Public key infrastructure (PKI) trustpoint	“Configuring and Managing a Cisco IOS Certificate Server for PKI Deployment” chapter in <i>Public Key Infrastructure Configuration Guide</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for X.509v3 Certificates for SSH Authentication

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 18: Feature Information for X.509v3 Certificates for SSH Authentication

Feature Information	Release	Modification
X.509v3 Certificates for SSH Authentication	Cisco IOS XE Denali 16.1.x	The X.509v3 Certificates for SSH Authentication feature uses the X.509v3 digital certificates in server and user authentication at the secure shell (SSH) server side



CHAPTER 10

Configuring Secure Socket Layer HTTP

- [Finding Feature Information, on page 157](#)
- [Information about Secure Socket Layer HTTP, on page 157](#)
- [How to Configure Secure Socket Layer HTTP, on page 161](#)
- [Monitoring Secure HTTP Server and Client Status, on page 167](#)
- [Additional References for Secure Socket Layer HTTP, on page 168](#)
- [Feature Information for Secure Socket Layer HTTP, on page 168](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information about Secure Socket Layer HTTP

Secure HTTP Servers and Clients Overview

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with `https://` instead of `http://`.



Note SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server

processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.



Note Beginning with Cisco IOS XE Denali 16.3.1, support for attaching IPv6 ACL to the HTTP server has been enabled. Prior to Cisco IOS XE Denali 16.3.1, only IPv4 ACL support was available for configuring the secure HTTP server. You can attach the preconfigured IPv6 and IPv4 ACLs to the HTTP server using the configuration CLI for the secure HTTP server.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you re-enable a secure HTTP connection.



Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

When a new certificate is enrolled, the new configuration change is not applied to the HTTPS server until the server is restarted. You can restart the server using either the CLI or by physical reboot. On restarting the server, the switch starts using the new certificate.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```

Device# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-3080755072
  revocation-check none
  rsakeypair TP-self-signed-3080755072
  !
!
crypto ca certificate chain TP-self-signed-3080755072
certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
  02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
  30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D

<output truncated>

```

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later re-enable a secure HTTP server, a new self-signed certificate is generated.



Note The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

For additional information on Certificate Authorities, see the “Configuring Certification Authority Interoperability” chapter in the *Cisco IOS Security Configuration Guide, Release 12.4*.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer Version 5.5 (or later) or Netscape Communicator Version 4.76 (or later). The `SSL_RSA_WITH_DES_CBC_SHA` CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. `SSL_RSA_WITH_DES_CBC_SHA`—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest

2. `SSL_RSA_WITH_NULL_SHA` key exchange with NULL for message encryption and SHA for message digest (only for SSL 3.0).
3. `SSL_RSA_WITH_NULL_MD5` key exchange with NULL for message encryption and MD5 for message digest (only for SSL 3.0).
4. `SSL_RSA_WITH_RC4_128_MD5`—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
5. `SSL_RSA_WITH_RC4_128_SHA`—RSA key exchange with RC4 128-bit encryption and SHA for message digest
6. `SSL_RSA_WITH_3DES_EDE_CBC_SHA`—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest
7. `SSL_RSA_WITH_AES_128_CBC_SHA`—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).
8. `SSL_RSA_WITH_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).
9. `SSL_RSA_WITH_DHE_AES_128_CBC_SHA`—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).
10. `SSL_RSA_WITH_DHE_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).



Note The latest versions of Chrome do not support the four original cipher suites, thus disallowing access to both web GUI and guest portals.

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

In a switch stack, the SSL session terminates at the active switch.

How to Configure Secure Socket Layer HTTP

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA Trustpoint:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	hostname <i>hostname</i> Example: Device (config)# hostname your_hostname	Specifies the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	ip domain-name <i>domain-name</i> Example: Device (config)# ip domain-name your_domain	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	crypto key generate rsa Example: Device (config)# crypto key generate rsa	(Optional) Generates an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
Step 5	crypto ca trustpoint <i>name</i> Example: Device (config)# crypto ca trustpoint your_trustpoint	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	enrollment url <i>url</i> Example:	Specifies the URL to which the switch should send certificate requests.

	Command or Action	Purpose
	<pre>Device(ca-trustpoint)# enrollment url http://your_server:80</pre>	
Step 7	<p>enrollment http-proxy <i>host-name</i> <i>port-number</i></p> <p>Example:</p> <pre>Device(ca-trustpoint)# enrollment http-proxy your_host 49</pre>	<p>(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server.</p> <ul style="list-style-type: none"> • For <i>host-name</i>, specify the proxy server used to get the CA. • For <i>port-number</i>, specify the port number used to access the CA.
Step 8	<p>crl query <i>url</i></p> <p>Example:</p> <pre>Device(ca-trustpoint)# crl query ldap://your_host:49</pre>	<p>Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.</p>
Step 9	<p>primary <i>name</i></p> <p>Example:</p> <pre>Device(ca-trustpoint)# primary your_trustpoint</pre>	<p>(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests.</p> <ul style="list-style-type: none"> • For <i>name</i>, specify the trustpoint that you just configured.
Step 10	<p>exit</p> <p>Example:</p> <pre>Device(ca-trustpoint)# exit</pre>	<p>Exits CA trustpoint configuration mode and return to global configuration mode.</p>
Step 11	<p>crypto ca authentication <i>name</i></p> <p>Example:</p> <pre>Device(config)# crypto ca authentication your_trustpoint</pre>	<p>Authenticates the CA by getting the public key of the CA. Use the same name used in Step 5.</p>
Step 12	<p>crypto ca enroll <i>name</i></p> <p>Example:</p> <pre>Device(config)# crypto ca enroll your_trustpoint</pre>	<p>Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.</p>
Step 13	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuring the Secure HTTP Server

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP server:

Before you begin

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

To verify the secure HTTP connection by using a Web browser, enter `https://URL`, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:



Note AES256_SHA2 is not supported.

```
https://209.165.129.1026
```

or

```
https://host.domain.com:1026
```

The existing **ip http access-class *access-list-number*** command for specifying the access-list (Only IPv4 ACLs) is going to be deprecated. You can still use this command to specify an access list to allow access to the HTTP server. Two new commands have been introduced to enable support for specifying IPv4 and IPv6 ACLs.

These are **ip http access-class *ipv4 access-list-name* | *access-list-number*** for specifying IPv4 ACLs and **ip http access-class *ipv6 access-list-name*** for specifying IPv6 ACLs. We recommend using the new CLI to avoid receiving warning messages.

Note the following considerations for specifying access-lists:

- If you specify an access-list that does not exist, the configuration takes place but you receive the below warning message:

```
ACL being attached does not exist, please configure it
```

- If you use the **ip http access-class** command for specifying an access-list for the HTTP server, the below warning message appears:

```
This CLI will be deprecated soon, Please use new CLI ip http
access-class ipv4/ipv6 <access-list-name>| <access-list-number>
```

- If you use **ip http access-class *ipv4 access-list-name* | *access-list-number*** or **ip http access-class *ipv6 access-list-name***, and an access-list was already configured using **ip http access-class**, the below warning message appears:

```
Removing ip http access-class <access-list-number>
```

ip http access-class *access-list-number* and **ip http access-class *ipv4 access-list-name* | *access-list-number*** share the same functionality. Each command overrides the configuration of the previous command. The

following combinations between the configuration of the two commands explain the effect on the running configuration:

- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-number* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-number* will be added to the running configuration.
- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-name* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-name* will be added to the running configuration.
- If **ip http access-class ipv4** *access-list-number* is already configured and you try to configure using **ip http access-class** *access-list-name*, the configuration of **ip http access-class ipv4** *access-list-number* will be removed from configuration and the configuration of **ip http access-class** *access-list-name* will be added to the running configuration.
- If **ip http access-class ipv4** *access-list-name* is already configured and you try to configure using **ip http access-class** *access-list-number*, the configuration of **ip http access-class ipv4** *access-list-name* will be removed from the configuration and the configuration of **ip http access-class** *access-list-number* will be added to the running configuration.

Procedure

	Command or Action	Purpose
Step 1	show ip http server status Example: <pre>Device# show ip http server status</pre>	(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output: <pre>HTTP secure server capability: Present</pre> or <pre>HTTP secure server capability: Not present</pre>
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip http secure-server Example: <pre>Device (config)# ip http secure-server</pre>	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.

	Command or Action	Purpose
Step 4	ip http secure-port <i>port-number</i> Example: Device(config)# ip http secure-port 443	(Optional) Specifies the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
Step 5	ip http secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: Device(config)# ip http secure-ciphersuite rc4-128-md5	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particularly CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	ip http secure-client-auth Example: Device(config)# ip http secure-client-auth	(Optional) Configures the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	ip http secure-trustpoint <i>name</i> Example: Device(config)# ip http secure-trustpoint your_trustpoint	Specifies the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. Note Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
Step 8	ip http path <i>path-name</i> Example: Device(config)# ip http path /your_server:80	(Optional) Sets a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
Step 9	ip http access-class <i>access-list-number</i> Example: Device(config)# ip http access-class 2	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 10	ip http access-class { ipv4 { <i>access-list-number</i> <i>access-list-name</i> } ipv6 { <i>access-list-name</i> } } Example:	(Optional) Specifies an access list to use to allow access to the HTTP server.

	Command or Action	Purpose
	Device (config) # ip http access-class ipv4 4	
Step 11	ip http max-connections <i>value</i> Example: Device (config) # ip http max-connections 4	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. We recommend that the value be at least 10 and not less. This is required for the UI to function as expected.
Step 12	ip http timeout-policy idle <i>seconds</i> life <i>seconds</i> requests <i>value</i> Example: Device (config) # ip http timeout-policy idle 120 life 240 requests 1	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances: <ul style="list-style-type: none"> • idle—the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • life—the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • requests—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 13	end Example: Device (config) # end	Returns to privileged EXEC mode.

Configuring the Secure HTTP Client

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP client:

Before you begin

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip http client secure-trustpoint <i>name</i> Example: Device(config)# <code>ip http client secure-trustpoint your_trustpoint</code>	(Optional) Specifies the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: Device(config)# <code>ip http client secure-ciphersuite rc4-128-md5</code>	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Monitoring Secure HTTP Server and Client Status

To monitor the SSL secure server and client status, use the privileged EXEC commands in the following table.

Table 19: Commands for Displaying the SSL Secure Server and Client Status

Command	Purpose
<code>show ip http client secure status</code>	Shows the HTTP secure client configuration.
<code>show ip http server secure status</code>	Shows the HTTP secure server configuration.
<code>show running-config</code>	Shows the generated self-signed certificate for secure HTTP connections.

Additional References for Secure Socket Layer HTTP

Related Documents

Related Topic	Document Title
Certification Authority	Configuring Certification Authority Interoperability

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Secure Socket Layer HTTP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 20: Feature Information for Secure Socket Layer HTTP

Feature Name	Releases	Feature Information
Secure Socket Layer HTTP		Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet.



CHAPTER 11

Configuring IPv4 ACLs

- [Finding Feature Information, on page 171](#)
- [Restrictions for Configuring IPv4 Access Control Lists, on page 171](#)
- [Information about Network Security with ACLs, on page 172](#)
- [How to Configure ACLs, on page 185](#)
- [Monitoring IPv4 ACLs, on page 204](#)
- [Configuration Examples for ACLs, on page 205](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Configuring IPv4 Access Control Lists

General Network Security

The following are restrictions for configuring network security with ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name. VLAN maps also accept a name.
- A standard ACL and an extended ACL cannot have the same name.
- Though visible in the command-line help strings, **appletalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- ACL wildcard is not supported in downstream client policy.
- Router ACL is enforced on all types of traffic, including CPU generated traffic.

- ACL logging in the egress direction are not supported for packets that are generated from the control plane of the device.
- If a downloadable ACL contains any type of duplicate entries, the entries are not auto merged. As a result, the 802.1X session authorization fails. Ensure that the downloadable ACL is optimized without any duplicate entries, for example port-based and name-based entries for the same port.

IPv4 ACL Network Interfaces

The following restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 2 interface that is a member of a VLAN, the Layer 2 (port) ACL takes precedence over an input Layer 3 ACL applied to the VLAN interface or a VLAN map applied to the VLAN.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic.
- If the **preauth_ipv4_acl** ACL is configured to filter packets, the ACL is removed after authentication.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

MAC ACLs on a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.



Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

Information about Network Security with ACLs

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

Access Control Entries

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

ACL Supported Types

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply port ACLs to a Layer 2 interface in each direction to each access list type — IPv4 and MAC.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).
- VLAN ACLs or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

Related Topics

[Restrictions for Configuring IPv4 Access Control Lists](#)

Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces. Port ACLs can be applied to the interface in outbound and inbound direction. The following access lists are supported:

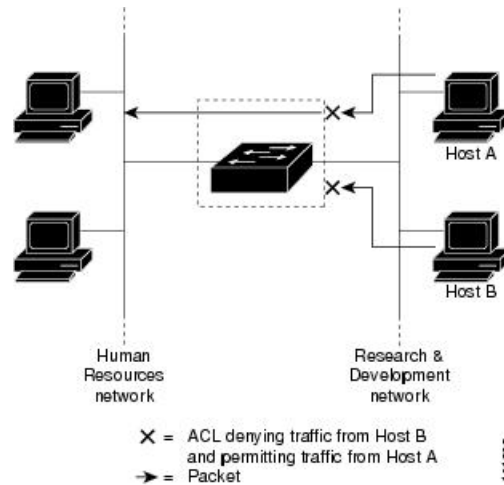
- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information
- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

Figure 6: Using ACLs to Control Traffic in a Network

This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but

prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the



inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.



Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network.

VLAN Maps

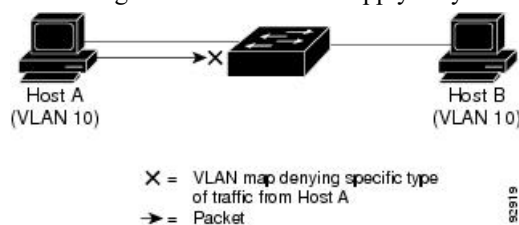
VLAN ACLs or VLAN maps are used to control network traffic within a VLAN. You can apply VLAN maps to all packets that are bridged within a VLAN in the switch or switch stack. VACLs are strictly for security packet filtering and for redirecting traffic to specific physical interfaces. VACLs are not defined by direction (ingress or egress).

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

Figure 7: Using VLAN Maps to Control Traffic

This figure shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.



ACEs and Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some access control entries (ACEs) do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.



Note For TCP ACEs with L4 Ops, the fragmented packets will be dropped per RFC 1858.

- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

ACEs and Fragmented and Unfragmented Traffic Examples

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
Device(config)# access-list 102 deny tcp any any
```



Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2, port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.

- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.

ACLs and Switch Stacks

ACL support is the same for a switch stack as for a standalone switch. ACL configuration information is propagated to all switches in the stack. All switches in the stack, including the active switch, process the information and program their hardware.

Active Switch and ACL Functions

The active switch performs these ACL functions:

- It processes the ACL configuration and propagates the information to all stack members.
- It distributes the ACL information to any switch that joins the stack.
- If packets must be forwarded by software for any reason (for example, not enough hardware resources), the active switch forwards the packets only after applying ACLs on the packets.
- It programs its hardware with the ACL information it processes.

Stack Member and ACL Functions

Stack members perform these ACL functions:

- They receive the ACL information from the active switch and program their hardware.
- A stack member configured as a standby switch, performs the functions of the active switch in the event the active switch fails.

Active Switch Failure and ACLs

Both the active and standby switches have the ACL information. When the active switch fails, the standby takes over. The new active switch distributes the ACL information to all stack members.

Standard and Extended IPv4 ACLs

This section describes IP ACLs.

An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

IPv4 ACL Switch Unsupported Features

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

The following ACL-related features are not supported:

- Non-IP protocol ACLs
- IP accounting
- Reflexive ACLs and dynamic ACLs are not supported.

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 21: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Numbered Standard IPv4 ACLs

When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with

non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to VLANs, to terminal lines, or to interfaces.

Numbered Extended IPv4 ACLs

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:



Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

These IP protocols are supported:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**)
- Internet Group Management Protocol (**igmp**)
- any Interior Protocol (**ip**)
- IP in IP tunneling (**ipinip**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**)
- Open Shortest Path First routing (**ospf**)
- Payload Compression Protocol (**pcp**)
- Protocol-Independent Multicast (**pim**)
- Transmission Control Protocol (**tcp**)
- User Datagram Protocol (**udp**)

Named IPv4 ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you

identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.



Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99 and . The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.
- A standard ACL and an extended ACL cannot have the same name.
- You can use standard or extended ACLs (named or numbered) in VLAN maps.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the **logging console** commands controlling the syslog messages.



Note ACL logging is not supported for ACLs used with Unicast Reverse Path Forwarding (uRPF). It is only supported for router ACL.



Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.



Note The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from crashing due to too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list.

Hardware and Software Treatment of IP ACLs

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.



Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch or stack member, then only the traffic in that VLAN arriving on that switch is affected.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the **log** keyword
- Generating ICMP unreachable messages

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show platform software fed switch { switch_num | active | standby } acl counters hardware** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Router ACLs function as follows:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If **log** has not been specified, the flows that match a *deny* statement in a security ACL are dropped by the hardware if *ip unreachables* is disabled. The flows matching a *permit* statement are switched in hardware.
- Adding the **log** keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a *permit* statement, the packet is still switched and routed in hardware.

VLAN Map Configuration Guidelines

VLAN maps are the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

The following are the VLAN map configuration guidelines:

- If there is no ACL configured to deny traffic on an interface and no VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- Logging is not supported for VLAN maps.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.

- If a VLAN map configuration cannot be applied in hardware, all packets in that VLAN are dropped.

VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.



Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have an router ACL and a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.
- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

```
permit... permit... permit... deny ip any any
```

or

```
deny... deny... deny... permit ip any any
```

- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

Time Ranges for ACLs

You can selectively apply extended ACLs based on the time of day and the week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables.

These are some benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the hardware memory. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)

**Note**

The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock.

Related Topics

[Configuring Time Ranges for ACLs](#), on page 193

IPv4 ACL Interface Considerations

When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Related Topics

[Applying an IPv4 ACL to an Interface](#), on page 196

[Restrictions for Configuring IPv4 Access Control Lists](#)

How to Configure ACLs

Configuring IPv4 ACLs

These are the steps to use IP ACLs on the switch:

Procedure

-
- Step 1** Create an ACL by specifying an access list number or name and the access conditions.
- Step 2** Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.
-

Creating a Numbered Standard ACL

Follow these steps to create a numbered standard ACL:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i> {deny permit} <i>source source-wildcard</i> [log] Example: Device(config)# access-list 2 deny <i>your_host</i>	Defines a standard IPv4 access list by using a source address and wildcard. The <i>access-list-number</i> is a decimal number from 1 to 99 or 1300 to 1999. Enter deny or permit to specify whether to deny or permit access if conditions are matched.

	Command or Action	Purpose
		<p>The <i>source</i> is the source address of the network or host from which the packet is being sent specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any as an abbreviation for <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword host as an abbreviation for <i>source</i> and <i>source-wildcard</i> of <i>source</i> 0.0.0.0. <p>(Optional) The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>(Optional) Enter log to cause an informational logging message about the packet that matches the entry to be sent to the console.</p> <p>Note Logging is supported only on ACLs attached to Layer 3 interfaces.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Related Topics

[Configuring VLAN Maps](#), on page 200

Creating a Numbered Extended ACL

Follow these steps to create a numbered extended ACL:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> {deny permit} protocol source source-wildcard destination destination-wildcard [precedence precedence] [tos tos] [fragments] [log [log-input] [time-range time-range-name] [dscp dscp] Example: Device(config)# access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log	<p>Defines an extended IPv4 access list and the access conditions.</p> <p>The <i>access-list-number</i> is a decimal number from 100 to 199 or 2000 to 2699.</p> <p>Enter deny or permit to specify whether to deny or permit the packet if conditions are matched.</p> <p>For <i>protocol</i>, enter the name or number of an P protocol: ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, pcp, pim, tcp, or udp, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword ip.</p> <p>Note This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.</p> <p>The <i>source</i> is the number of the network or host from which the packet is sent.</p> <p>The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>The <i>destination</i> is the network or host number to which the packet is sent.</p> <p>The <i>destination-wildcard</i> applies wildcard bits to the destination.</p> <p>Source, source-wildcard, destination, and destination-wildcard can be specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any for 0.0.0.0 255.255.255.255 (any host).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • The keyword host for a single host 0.0.0.0. <p>The other keywords are optional and have these meanings:</p> <ul style="list-style-type: none"> • precedence—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: routine (0), priority (1), immediate (2), flash (3), flash-override (4), critical (5), internet (6), network (7). • fragments—Enter to check non-initial fragments. • tos—Enter to match by type of service level, specified by a number from 0 to 15 or a name: normal (0), max-reliability (2), max-throughput (4), min-delay (8). • log—Enter to create an informational logging message to be sent to the console about the packet that matches the entry or log-input to include the input interface in the log entry. • time-range—Specify the time-range name. • dscp—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values. <p>Note If you enter a dscp value, you cannot enter tos or precedence. You can enter both a tos and a precedence value with no dscp.</p>
Step 3	<p>access-list <i>access-list-number</i> {deny permit} tcp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [established] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>] [<i>flag</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit tcp any any eq 500</pre>	<p>Defines an extended TCP access list and the access conditions.</p> <p>The parameters are the same as those described for an extended IPv4 ACL, with these exceptions:</p> <p>(Optional) Enter an <i>operator</i> and <i>port</i> to compare source (if positioned after <i>source source-wildcard</i>) or destination (if positioned after <i>destination destination-wildcard</i>) port. Possible operators include eq (equal), gt (greater than), lt (less than), neq (not equal), and range (inclusive range). Operators require</p>

	Command or Action	Purpose
		<p>a port number (range requires two port numbers separated by a space).</p> <p>Enter the <i>port</i> number as a decimal number (from 0 to 65535) or the name of a TCP port. Use only TCP port numbers or names when filtering TCP.</p> <p>The other optional keywords have these meanings:</p> <ul style="list-style-type: none"> • established—Enter to match an established connection. This has the same function as matching on the ack or rst flag. • <i>flag</i>—Enter one of these flags to match by the specified TCP header bits: ack (acknowledge), fin (finish), psh (push), rst (reset), syn (synchronize), or urg (urgent).
Step 4	<p>access-list <i>access-list-number</i> {deny permit} udp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit udp any any eq 100</pre>	<p>(Optional) Defines an extended UDP access list and the access conditions.</p> <p>The UDP parameters are the same as those described for TCP except that the [<i>operator port</i>] port number or name must be a UDP port number or name, and the flag and established keywords are not valid for UDP.</p>
Step 5	<p>access-list <i>access-list-number</i> {deny permit} icmp <i>source source-wildcard destination</i> <i>destination-wildcard</i> [<i>icmp-type</i> [<i>icmp-type</i> <i>icmp-code</i>] [<i>icmp-message</i>]] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit icmp any any 200</pre>	<p>Defines an extended ICMP access list and the access conditions.</p> <p>The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <i>icmp-type</i>—Enter to filter by ICMP message type, a number from 0 to 255. • <i>icmp-code</i>—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • <i>icmp-message</i>—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.

	Command or Action	Purpose
Step 6	<p>access-list <i>access-list-number</i> {deny permit} igmp <i>source source-wildcard destination destination-wildcard</i> [<i>igmp-type</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit igmp any any 14</pre>	<p>(Optional) Defines an extended IGMP access list and the access conditions.</p> <p>The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.</p> <p><i>igmp-type</i>—To match IGMP message type, enter a number from 0 to 15, or enter the message name: dvmrp, host-query, host-report, pim, or trace.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Related Topics

[Configuring VLAN Maps](#), on page 200

Creating Named Standard ACLs

Follow these steps to create a standard ACL using names:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>ip access-list standard <i>name</i></p> <p>Example:</p> <pre>Device(config)# ip access-list standard 20</pre>	<p>Defines a standard IPv4 access list using a name, and enter access-list configuration mode.</p> <p>The name can be a number from 1 to 99.</p>

	Command or Action	Purpose
Step 4	<p>Use one of the following:</p> <ul style="list-style-type: none"> • deny {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] • permit {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] <p>Example:</p> <pre>Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255</pre> <p>or</p> <pre>Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0</pre>	<p>In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped.</p> <ul style="list-style-type: none"> • host <i>source</i>—A source and source wildcard of <i>source</i> 0.0.0.0. • any—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-std-nacl)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating Extended Named ACLs

Follow these steps to create an extended ACL using names:

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended name Example: Device(config)# ip access-list extended 150	Defines an extended IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 100 to 199.
Step 4	{deny permit} protocol {source [source-wildcard] host source any} {destination [destination-wildcard] host destination any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name] Example: Device(config-ext-nacl)# permit 0 any any	In access-list configuration mode, specify the conditions allowed or denied. Use the log keyword to get access list logging messages, including violations. <ul style="list-style-type: none"> • host source—A source and source wildcard of <i>source</i> 0.0.0.0. • host destination—A destination and destination wildcard of <i>destination</i> 0.0.0.0. • any—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 5	end Example: Device(config-ext-nacl)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

When you are creating extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After creating a named ACL, you can apply it to interfaces or to VLANs .

Configuring Time Ranges for ACLs

Follow these steps to configure a time-range parameter for an ACL:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device(config)# enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	time-range <i>time-range-name</i> Example: Device(config)# time-range workhours	Assigns a meaningful name (for example, <i>workhours</i>) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 4	Use one of the following: <ul style="list-style-type: none"> • absolute [<i>start time date</i>] [<i>end time date</i>] • periodic <i>day-of-the-week hh:mm to [day-of-the-week] hh:mm</i> • periodic {<i>weekdays</i> <i>weekend</i> daily} <i>hh:mm to hh:mm</i> Example: Device(config-time-range)# absolute start	Specifies when the function it will be applied to is operational. <ul style="list-style-type: none"> • You can use only one absolute statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple periodic statements. For example, you could configure different hours for weekdays and weekends.

	Command or Action	Purpose
	<pre>00:00 1 Jan 2006 end 23:59 1 Jan 2006</pre> <p>or</p> <pre>Device(config-time-range)# periodic weekdays 8:00 to 12:00</pre>	See the example configurations.
Step 5	<pre>end</pre> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<pre>show running-config</pre> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	<pre>copy running-config startup-config</pre> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Repeat the steps if you have multiple items that you want in effect at different times.

Related Topics

[Time Ranges for ACLs](#), on page 184

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

Follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

Procedure

	Command or Action	Purpose
Step 1	<pre>enable</pre> <p>Example:</p>	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device(config)# enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	line [console vty] line-number Example: Device(config)# line console 0	Identifies a specific line to configure, and enter in-line configuration mode. <ul style="list-style-type: none"> • console—Specifies the console terminal line. The console port is DCE. • vtty—Specifies a virtual terminal for remote console access. <p>The <i>line-number</i> is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.</p>
Step 4	access-class access-list-number {in out} Example: Device(config-line)# access-class 10 in	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 5	end Example: Device(config-line)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Applying an IPv4 ACL to an Interface

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Identifies a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	ip access-group {<i>access-list-number</i> <i>name</i>} {in out} Example: Device(config-if)# ip access-group 2 in	Controls access to the specified interface.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Displays the access list configuration.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[IPv4 ACL Interface Considerations](#), on page 184

Restrictions for Configuring IPv4 Access Control Lists

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Follow these steps to create a named MAC extended ACL:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mac access-list extended name Example: Device(config)# mac access-list extended macl	Defines an extended MAC access list using a name.
Step 4	{deny permit} {any host source MAC address source MAC address mask} {any host destination MAC address destination MAC address mask} [type mask lsap lsap mask aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat lavc-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp 0-65535] [cos cos] Example: Device(config-ext-macl)# deny any any decnet-iv or Device(config-ext-macl)# permit any any	In extended MAC access-list configuration mode, specifies to permit or deny any source MAC address, a source MAC address with a mask, or a specific host source MAC address and any destination MAC address, destination MAC address with a mask, or a specific destination MAC address. (Optional) You can also enter these options: <ul style="list-style-type: none"> • <i>type mask</i>—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits applied to the EtherType before testing for a match. • <i>lsap lsap mask</i>—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat lavc-sca mop-console mop-dump msdos mumps nethbios vines-echo vines-ip xns-idp—A non-IP protocol. • cos <i>cos</i>—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 5	end Example: Device(config-ext-macl) # end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Restrictions for Configuring IPv4 Access Control Lists](#)

[Configuring VLAN Maps](#), on page 200

Applying a MAC ACL to a Layer 2 Interface

Follow these steps to apply a MAC access list to control access to a Layer 2 interface:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Identifies a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 4	mac access-group {<i>name</i>} {in out } Example: Device(config-if)# mac access-group mac1 in	Controls access to the specified interface by using the MAC access list. Port ACLs are supported in the outbound and inbound directions .
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show mac access-group [interface <i>interface-id</i>] Example: Device# show mac access-group interface gigabitethernet1/0/2	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an

undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Related Topics

[Restrictions for Configuring IPv4 Access Control Lists](#)

Configuring VLAN Maps

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before you begin

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN.

Procedure

	Command or Action	Purpose
Step 1	vlan access-map <i>name</i> [<i>number</i>] Example: Device(config)# vlan access-map map_1 20	Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map. When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete. VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match. Entering this command changes to access-map configuration mode.
Step 2	match { <i>ip</i> <i>mac</i> } address { <i>name</i> <i>number</i> } [<i>name</i> <i>number</i>] Example: Device(config-access-map)# match ip address ip2	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.

	Command or Action	Purpose
		<p>Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.</p>
Step 3	<p>Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):</p> <ul style="list-style-type: none"> • action { forward } <pre>Device(config-access-map) # action forward</pre> <ul style="list-style-type: none"> • action { drop } <pre>Device(config-access-map) # action drop</pre>	Sets the action for the map entry.
Step 4	<p>vlan filter <i>mapname</i> vlan-list <i>list</i></p> <p>Example:</p> <pre>Device(config) # vlan filter map 1 vlan-list 20-22</pre>	<p>Applies the VLAN map to one or more VLAN IDs.</p> <p>The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.</p>

Related Topics

- [Creating a Numbered Standard ACL](#), on page 185
- [Creating a Numbered Extended ACL](#), on page 187
- [Creating Named MAC Extended ACLs](#), on page 197
- [Creating a VLAN Map](#), on page 201
- [Applying a VLAN Map to a VLAN](#), on page 203

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vlan access-map <i>name</i> [<i>number</i>] Example: Device(config)# vlan access-map map_1 20	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 3	match {ip mac} address {<i>name</i> <i>number</i>} [<i>name</i> <i>number</i>] Example: Device(config-access-map)# match ip address ip2	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 4	action {drop forward} Example: Device(config-access-map)# action forward	(Optional) Sets the action for the map entry. The default is to forward.
Step 5	end Example: Device(config-access-map)# end	Returns to global configuration mode.
Step 6	show running-config Example:	Displays the access list configuration.

	Command or Action	Purpose
	Device# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Related Topics

[Configuring VLAN Maps](#), on page 200

Applying a VLAN Map to a VLAN

To apply a VLAN map to one or more VLANs, perform these steps.

Procedure

	Command or Action	Purpose
Step 1		
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vlan filter <i>mapname</i> vlan-list <i>list</i> Example: Device(config)# <code>vlan filter map 1 vlan-list 20-22</code>	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Displays the access list configuration.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Configuring VLAN Maps](#), on page 200

Monitoring IPv4 ACLs

You can monitor IPv4 ACLs by displaying the ACLs that are configured on the switch, and displaying the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Table 22: Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [<i>number</i> <i>name</i>]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [<i>number</i> <i>name</i>]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface <i>interface-id</i>	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the ip access-group interface configuration command, the access groups are included in the display.
show running-config [interface <i>interface-id</i>]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [interface <i>interface-id</i>]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

Configuration Examples for ACLs

Examples: Using Time Ranges with ACLs

This example shows how to verify after you configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday.

```
Device# show time-range
time-range entry: new_year_day_2003 (inactive)
  absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
  periodic weekdays 8:00 to 12:00
  periodic weekdays 13:00 to 17:00
```

To apply a time range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Device(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Device(config)# access-list 188 permit tcp any any time-range workhours
Device(config)# end
Device# show access-lists
Extended IP access list 188
  10 deny tcp any any time-range new_year_day_2006 (inactive)
  20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Device(config)# ip access-list extended deny_access
Device(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Device(config-ext-nacl)# exit
Device(config)# ip access-list extended may_access
Device(config-ext-nacl)# permit tcp any any time-range workhours
Device(config-ext-nacl)# end
Device# show ip access-lists
Extended IP access list lpip_default
  10 permit ip any any
Extended IP access list deny_access
  10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
  10 permit tcp any any time-range workhours (inactive)
```

Examples: Including Comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would

be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith through
Device(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Device(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

IPv4 ACL Configuration Examples

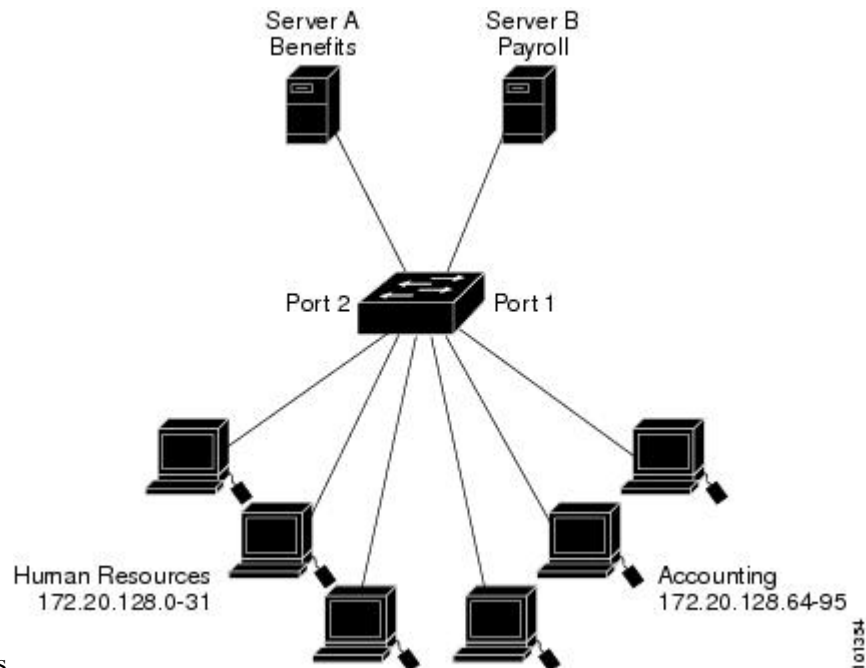
This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.4* and to the “Configuring IP Services” section in the “IP Addressing and Services” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

ACLs in a Small Networked Office

Figure 8: Using Router ACLs to Control Traffic

This shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing

confidential payroll data. All users can access Server A, but Server B has restricted



access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Examples: ACLs in a Small Networked Office

This example uses a standard ACL to filter traffic coming into Server B from a port, permitting traffic only from Accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic coming out of routed Port 1 from the specified source address.

```
Device(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Device(config)# end
Device# show access-lists
Standard IP access list 6
 10 permit 172.20.128.64, wildcard bits 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic coming from Server B into a port, permitting traffic from any source address (in this case Server B) to only the Accounting destination addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. Note that with extended ACLs, you must enter the protocol (IP) before the source and destination information.

```
Device(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Device(config)# end
Device# show access-lists
Extended IP access list 106
```

```

10 permit ip any 172.20.128.64 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 106 in

```

Example: Numbered ACLs

In this example, network 36.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 36.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 36.0.0.0 subnets. The ACL is applied to packets entering a port.

```

Device(config)# access-list 2 permit 36.48.0.3
Device(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 2 in

```

Examples: Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```

Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# access-list 102 permit icmp any any
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 102 in

```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

```

Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in

```

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 on stack member 1 is the interface that connects the router to the Internet.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in
```

Examples: Named ACLs

Creating named standard and extended ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 1.2.3.4
Device(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Device(config)# interface gigabitethernet3/0/2
Device(config-if)# no switchport
Device(config-if)# ip address 2.0.5.1 255.255.255.0
Device(config-if)# ip access-group Internet_filter out
Device(config-if)# ip access-group marketing_group in
```

Deleting individual ACEs from named ACLs

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Examples: Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m. (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```

Device(config)# time-range no-http
Device(config)# periodic weekdays 8:00 to 18:00
!
Device(config)# time-range udp-yes
Device(config)# periodic weekend 12:00 to 20:00
!
Device(config)# ip access-list extended strict
Device(config-ext-nacl)# deny tcp any any eq www time-range no-http
Device(config-ext-nacl)# permit udp any any time-range udp-yes
!
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group strict in

```

Examples: Configuring Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```

Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith workstation through
Device(config)# access-list 1 deny 171.69.3.13

```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```

Device(config)# access-list 100 remark Do not allow Winter to browse the web
Device(config)# access-list 100 deny host 171.69.3.85 any eq www
Device(config)# access-list 100 remark Do not allow Smith to browse the web
Device(config)# access-list 100 deny host 171.69.3.13 any eq www

```

In this example of a named ACL, the Jones subnet is not allowed access:

```

Device(config)# ip access-list standard prevention
Device(config-std-nacl)# remark Do not allow Jones subnet through
Device(config-std-nacl)# deny 171.69.0.0 0.0.255.255

```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```

Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Device(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet

```

Examples: ACL Logging

Two variations of logging are supported on router ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.


```

Device(config)# ip access-list standard stan1
Device(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Device(config-std-nacl)# permit any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group stan1 in
Device(config-if)# end
Device# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: level debugging, 37 messages logged
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 37 messages logged
  File logging: disabled
  Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet

```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```

Device(config)# ip access-list extended ext1
Device(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Device(config-ext-nacl)# deny udp any any log
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip access-group ext1 in

```

This is an example of a log for an extended ACL:

```

01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8 packets

```

Note that all logging entries for IP ACLs start with %SEC-6-IPACCESSLOG with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```

00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
->
10.1.1.61 (0/0), 1 packet

```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

Configuration Examples for ACLs and VLAN Maps

Example: Creating an ACL and a VLAN Map to Deny a Packet

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1* ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Device(config)# ip access-list extended ip1
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 10
Device(config-access-map)# match ip address ip1
Device(config-access-map)# action drop
```

Example: Creating an ACL and a VLAN Map to Permit a Packet

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Device(config)# ip access-list extended ip2
Device(config-ext-nacl)# permit udp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 20
Device(config-access-map)# match ip address ip2
Device(config-access-map)# action forward
```

Example: Default Action of Dropping IP Packets and Forwarding MAC Packets

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets
- Forward all non-IP packets

```
Device(config)# access-list 101 permit udp any any
Device(config)# ip access-list extended igmp-match
Device(config-ext-nacl)# permit igmp any any

Device(config)# action forward
```

```

Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-ip-default 10
Device(config-access-map)# match ip address 101
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 20
Device(config-access-map)# match ip address igmp-match
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 30
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward

```

Example: Default Action of Dropping MAC Packets and Forwarding IP Packets

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

```

Device(config)# mac access-list extended good-hosts
Device(config-ext-nacl)# permit host 000.0c00.0111 any
Device(config-ext-nacl)# permit host 000.0c00.0211 any
Device(config-ext-nacl)# exit
Device(config)# action forward
Device(config-ext-nacl)# mac access-list extended good-protocols
Device(config-ext-nacl)# permit any any vines-ip
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-mac-default 10
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-mac-default 20
Device(config-access-map)# match mac address good-protocols
Device(config-access-map)# action forward

```

Example: Default Action of Dropping All Packets

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```

Device(config)# vlan access-map drop-all-default 10

```

```

Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-all-default 20
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward

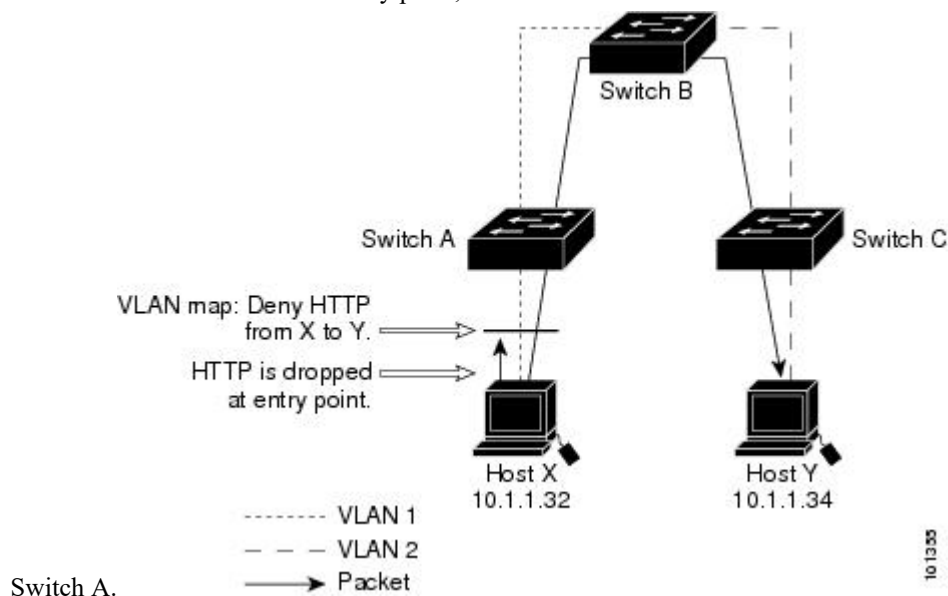
```

Configuration Examples for Using VLAN Maps in Your Network

Example: Wiring Closet Configuration

Figure 9: Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. Assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point,



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```

Device(config)# ip access-list extended http
Device(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Device(config-ext-nacl)# exit

```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```

Device(config)# vlan access-map map2 10
Device(config-access-map)# match ip address http

```

```

Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# ip access-list extended match_all
Device(config-ext-nacl)# permit ip any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map2 20
Device(config-access-map)# match ip address match_all
Device(config-access-map)# action forward

```

Then, apply VLAN access map *map2* to VLAN 1.

```

Device(config)# vlan filter map2 vlan 1

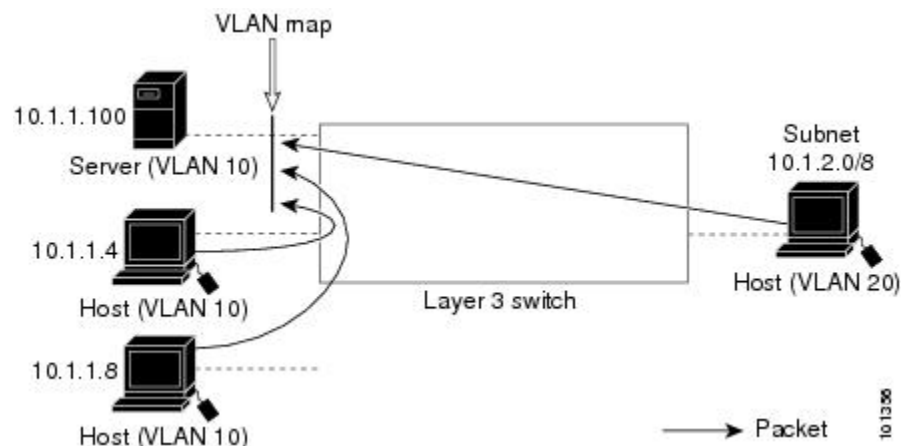
```

Example: Restricting Access to a Server on Another VLAN

Figure 10: Restricting Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts:

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.



Example: Denying Access to a Server on Another VLAN

This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Define the IP ACL that will match the correct packets.

```

Device(config)# ip access-list extended SERVER1_ACL
Device(config-ext-nacl)# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.4 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.8 host 10.1.1.100
Device(config-ext-nacl)# exit

```

Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```
Device(config)# vlan access-map SERVER1_MAP
Device(config-access-map)# match ip address SERVER1_ACL
Device(config-access-map)# action drop
Device(config)# vlan access-map SERVER1_MAP 20
Device(config-access-map)# action forward
Device(config-access-map)# exit
```

Apply the VLAN map to VLAN 10.

```
Device(config)# vlan filter SERVER1_MAP vlan-list 10
```

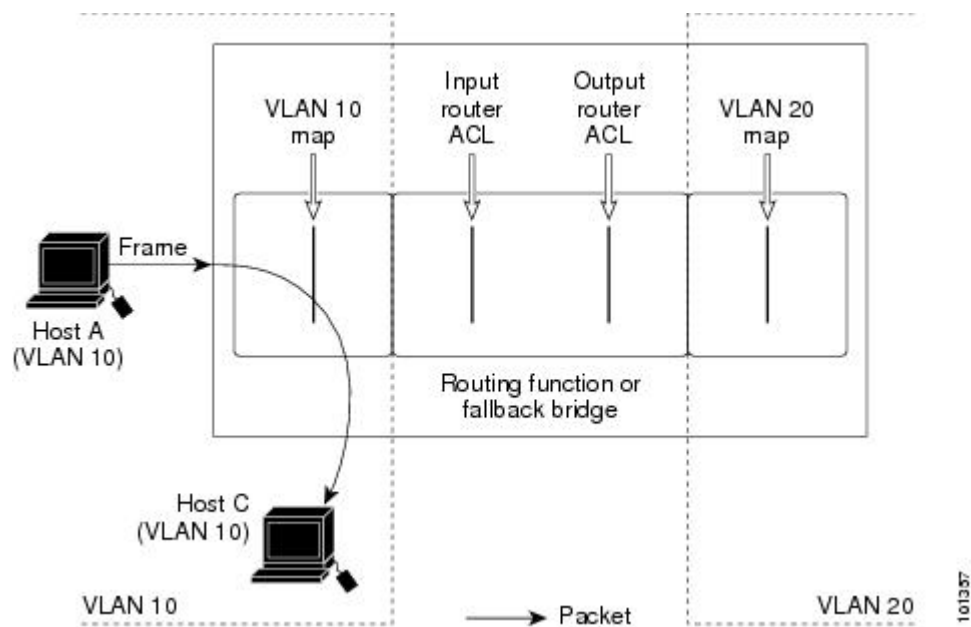
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

Example: ACLs and Switched Packets

Figure 11: Applying ACLs on Switched Packets

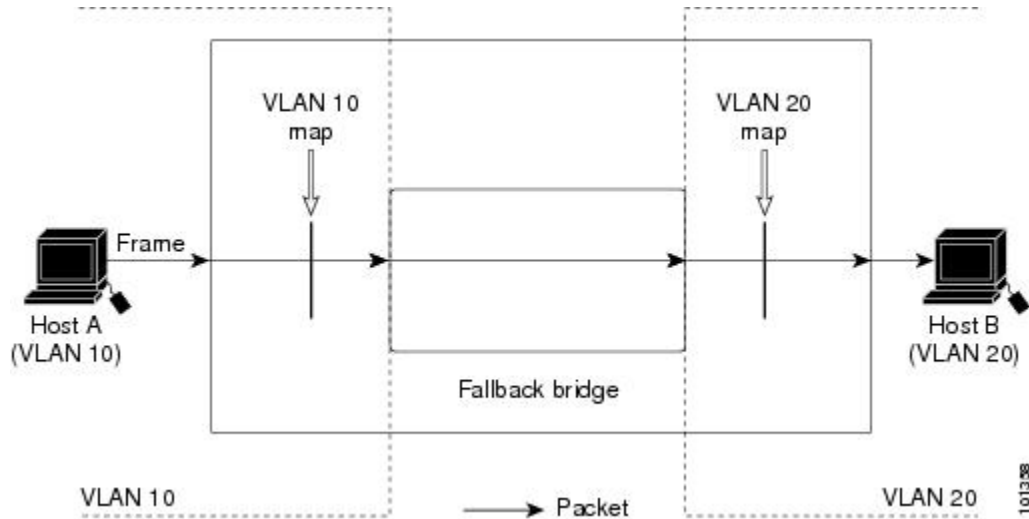
This example shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.



Example: ACLs and Bridged Packets

Figure 12: Applying ACLs on Bridged Packets

This example shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.

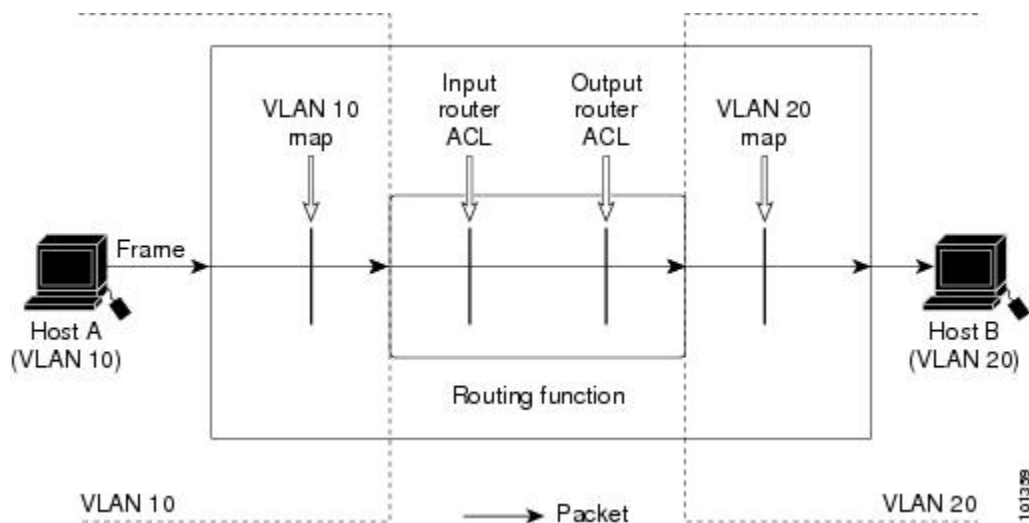


Example: ACLs and Routed Packets

Figure 13: Applying ACLs on Routed Packets

This example shows how ACLs are applied on routed packets. The ACLs are applied in this order:

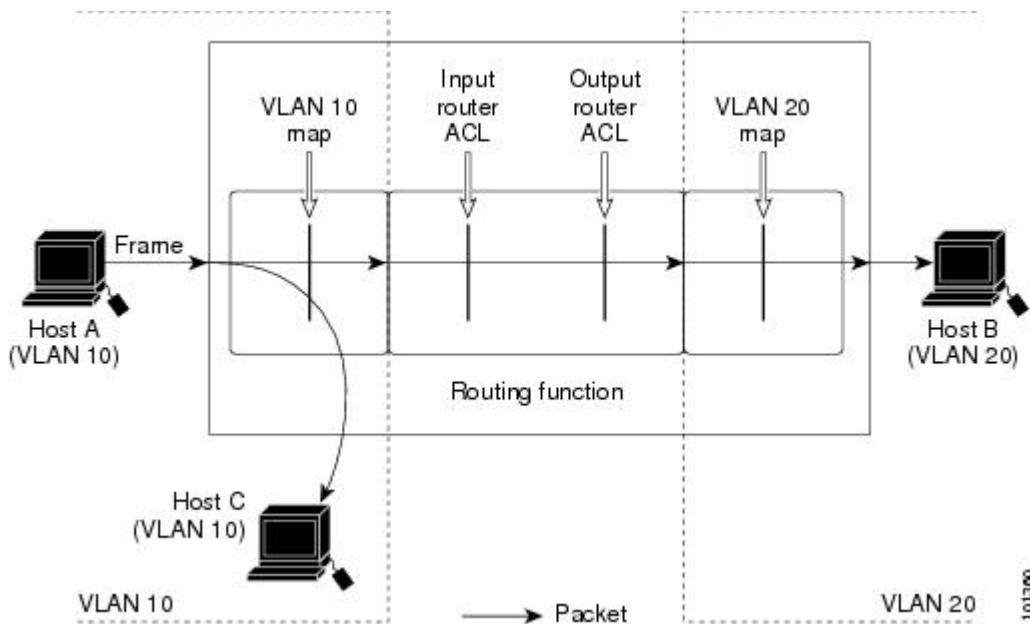
1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN



Example: ACLs and Multicast Packets

Figure 14: Applying ACLs on Multicast Packets

This example shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN. The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map drops the packet, no destination receives a copy of the packet.





CHAPTER 12

Configuring IPv6 ACLs

- Finding Feature Information, on page 219
- IPv6 ACLs Overview, on page 219
- Restrictions for IPv6 ACLs, on page 221
- Default Configuration for IPv6 ACLs , on page 222
- Configuring IPv6 ACLs, on page 222
- Attaching an IPv6 ACL to an Interface, on page 226
- Configuring a VLAN Map, on page 227
- Applying a VLAN Map to a VLAN, on page 229
- Monitoring IPv6 ACLs, on page 230

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

IPv6 ACLs Overview

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similar to how you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running IP base and LAN base feature sets.

A switch supports three types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on outbound and inbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.

- VLAN ACLs or VLAN maps access-control all packets in a VLAN. You can use VLAN maps to filter traffic between devices in the same VLAN. ACL VLAN maps are applied on L2 VLANs. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv6. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map.

The switch supports VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs.

Switch Stacks and IPv6 ACLs

The active switch supports IPv6 ACLs in hardware and distributes the IPv6 ACLs to the stack members.

If a standby switch takes over as the active switch, it distributes the ACL configuration to all stack members. The member switches sync up the configuration distributed by the new active switch and flush out entries that are not required.

When an ACL is modified, attached to, or detached from an interface, the active switch distributes the change to all stack members.

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

Related Topics

[Restrictions for Configuring IPv4 Access Control Lists](#)

VLAN Maps

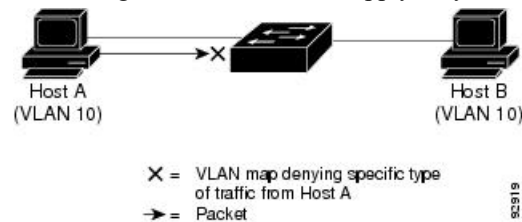
VLAN ACLs or VLAN maps are used to control network traffic within a VLAN. You can apply VLAN maps to all packets that are bridged within a VLAN in the switch or switch stack. VACLs are strictly for security packet filtering and for redirecting traffic to specific physical interfaces. VACLs are not defined by direction (ingress or egress).

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

Figure 15: Using VLAN Maps to Control Traffic

This figure shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.



Interactions with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, packets are dropped on the interface and an unload error message is logged.

Restrictions for IPv6 ACLs

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- The switch does not support matching on these keywords: **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).
- This release supports port ACLs, router ACLs and VLAN ACLs (VLAN maps) for IPv6.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface.

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, the packets associated with the ACL are dropped on the interface.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.

Default Configuration for IPv6 ACLs

The default IPv6 ACL configuration is as follows:

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

Configuring IPv6 ACLs

To filter IPv6 traffic, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	[no]{ipv6 access-list list-name client permit-control-packets log-update threshold role-based list-name} Example: Device (config)# ipv6 access-list example_acl_list	Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.
Step 4	[no]{deny permit} protocol {source-ipv6-prefix/ prefix-length any threshold host source-ipv6-address} [operator [port-number]] { destination-ipv6-prefix/ prefix-length any host destination-ipv6-address} [operator [port-number]][dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: <ul style="list-style-type: none"> • For protocol, enter the name or number of an IP: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/ prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix <code>::/0</code>. • For host <i>source-ipv6-address</i> or <i>destination-ipv6-address</i>, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol.

	Command or Action	Purpose
		<p>Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range.</p> <p>If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6- prefix/prefix-length</i> argument, it must match the destination port.</p> <ul style="list-style-type: none"> • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4,294,967,295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
<p>Step 5</p>	<p>{deny permit} tcp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator</p>	<p>(Optional) Define a TCP access list and the access conditions.</p>

	Command or Action	Purpose
	<pre>[port-number]] {destination-ipv6- prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neq {port protocol}] [psh] [range {port protocol}] [rst] [routing] [sequence value] [syn] [time-range name] [urg]</pre>	<p>Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters:</p> <ul style="list-style-type: none"> • ack: Acknowledgment bit set. • established: An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • fin: Finished bit set; no more data from sender. • neq {port protocol}: Matches only packets that are not on a given port number. • psh—Push function bit set. • range {port protocol}: Matches only packets in the port number range. • rst: Reset bit set. • syn: Synchronize bit set. • urg: Urgent pointer bit set.
Step 6	<pre>{deny permit} udp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [log] [log-input] [neq {port protocol}] [range {port protocol}] [routing] [sequence value] [time-range name]]</pre>	<p>(Optional) Define a UDP access list and the access conditions.</p> <p>Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [operator [port]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.</p>
Step 7	<pre>{deny permit} icmp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [icmp-type [icmp-code] icmp-message] [dscp value] [log] [log-input] [routing] [sequence value] [time-range name]</pre>	<p>(Optional) Define an ICMP access list and the access conditions.</p> <p>Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 1, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • icmp-type: Enter to filter by ICMP message type, a number from 0 to 255. • icmp-code: Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255.

	Command or Action	Purpose
		<ul style="list-style-type: none"> <i>icmp-message</i>: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 8	end	Return to privileged EXEC mode.
Step 9	show ipv6 access-list	Verify the access list configuration.
Step 10	show running-config Example: Device# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Attaching an IPv6 ACL to an Interface

You can apply an ACL to outbound or inbound traffic on Layer 3 interfaces, or to inbound traffic on Layer 2 interfaces. You can also apply ACLs only to inbound management traffic on Layer 3 interfaces.

Follow these steps to control access to an interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<code>interface <i>interface-id</i></code>	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
Step 4	<code>no switchport</code>	If applying a router ACL, this changes the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 5	<code>ipv6 address <i>ipv6-address</i></code>	Configure an IPv6 address on a Layer 3 interface (for router ACLs).
Step 6	<code>ipv6 traffic-filter <i>access-list-name</i> {in out}</code>	Apply the access list to incoming or outgoing traffic on the interface.
Step 7	<code>end</code> Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 8	<code>show running-config</code> Example: Device# <code>show running-config</code>	Verifies your entries.
Step 9	<code>copy running-config startup-config</code> Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a VLAN Map

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before you begin

Create the IPv6 ACL that you want to apply to the VLAN.

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code> Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vlan access-map <i>name</i> [<i>number</i>] Example: Device(config)# vlan access-map map_1 20	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 4	match { ip ipv6 mac } address { <i>name</i> <i>number</i> } [<i>name</i> <i>number</i>] Example: Device(config-access-map)# match ipv6 address ip_net	<p>Match the packet against one or more access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against IP access lists. Non-IP packets are only matched against named MAC access lists.</p> <p>Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.</p>
Step 5	Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs: <ul style="list-style-type: none"> • action { forward } 	Sets the action for the map entry.

	Command or Action	Purpose
	<pre>Device(config-access-map)# action forward</pre> <ul style="list-style-type: none"> • action { drop} <pre>Device(config-access-map)# action drop</pre>	
Step 6	<p>vlan filter <i>mapname</i> vlan-list <i>list</i></p> <p>Example:</p> <pre>Device(config)# vlan filter map 1 vlan-list 20-22</pre>	<p>Applies the VLAN map to one or more VLAN IDs.</p> <p>The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.</p>

Applying a VLAN Map to a VLAN

To apply a VLAN map to one or more VLANs, perform these steps.

Procedure

	Command or Action	Purpose
Step 1		
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>vlan filter <i>mapname</i> vlan-list <i>list</i></p> <p>Example:</p> <pre>Device(config)# vlan filter map 1 vlan-list 20-22</pre>	<p>Applies the VLAN map to one or more VLAN IDs.</p> <p>The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Displays the access list configuration.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Related Topics

[Configuring VLAN Maps](#), on page 200

Monitoring IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the privileged EXEC commands shown in the table below:

Table 23: show ACL commands

Command	Purpose
show access-lists	Displays all access lists configured on the switch.
show ipv6 access-list [<i>access-list-name</i>]	Displays all configured IPv6 access lists or the access list specified by name.
show vlan access-map [<i>map-name</i>]	Displays VLAN access map configuration.
show vlan filter [access-map <i>access-map</i> vlan <i>vlan-id</i>]	Displays the mapping between VACLs and VLANs.

This is an example of the output from the `show access-lists` privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch # show access-lists
Extended IP access list hello
  10 permit ip any any
IPv6 access list ipv6
  permit ipv6 any any sequence 10
```

This is an example of the output from the `show ipv6 access-list` privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack

```
Switch# show ipv6 access-list
IPv6 access list inbound
  permit tcp any any eq bgp (8 matches) sequence 10
  permit tcp any any eq telnet (15 matches) sequence 20
  permit udp any any sequence 30
IPv6 access list outbound
  deny udp any any sequence 10
  deny tcp any any eq telnet sequence 20
```

This is an example of the output from the `show vlan access-map` privileged EXEC command. The output shows VLAN access map information.

```
Switch# show vlan access-map
Vlan access-map "m1" 10
  Match clauses:
    ipv6 address: ip2
  Action: drop
```




CHAPTER 13

Configuring DHCP

- [Finding Feature Information, on page 233](#)
- [Information About DHCP, on page 233](#)
- [How to Configure DHCP Features, on page 240](#)
- [Configuring DHCP Server Port-Based Address Allocation, on page 246](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About DHCP

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator. The switch can act as a DHCP server.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on output interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.



Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received through an untrusted interface. By default, the switch considers all interfaces untrusted. So, the switch must be configured to trust some interfaces to use DHCP Snooping. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.

In a service-provider network, an example of an interface you might configure as trusted is one connected to a port on a device in the same network. An example of an untrusted interface is one that is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCP OFFER, DHCP ACK, DHCP NAK, or DHCP REQUEST packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCP RELEASE or DHCP DECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.
- The maximum snooping queue size of 1000 is exceeded when DHCP snooping is enabled.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on untrusted input interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

Normally, it is not desirable to broadcast packets to wireless clients. So, DHCP snooping replaces destination broadcast MAC address (ffff.ffff.ffff) with unicast MAC address for DHCP packets that are going from server to wireless clients. The unicast MAC address is retrieved from CHADDR field in the DHCP payload. This processing is applied for server to client packets such as DHCP OFFER, DHCP ACK, and DHCP NACK messages. The **ip dhcp snooping wireless bootp-broadcast enable** can be used to revert this behavior. When the wireless BOOTP broadcast is enabled, the broadcast DHCP packets from server are forwarded to wireless clients without changing the destination MAC address.

Option-82 Data Insertion

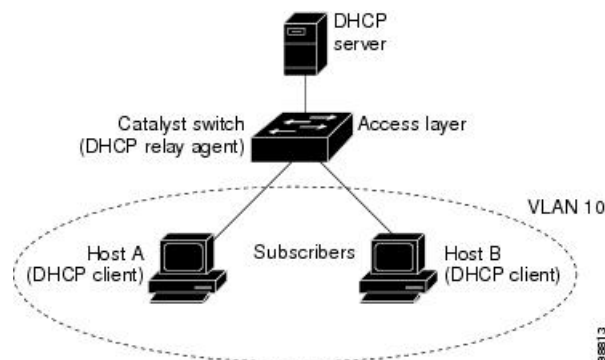
In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.



Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled on the VLANs to which subscriber devices using option-82 are assigned.

The following illustration shows a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 16: DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, the following sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.

- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received. You can configure the remote ID and circuit ID.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

In the default suboption configuration, when the described sequence of events occurs, the values in these fields do not change (see the illustration, *Suboption Packet Formats*):

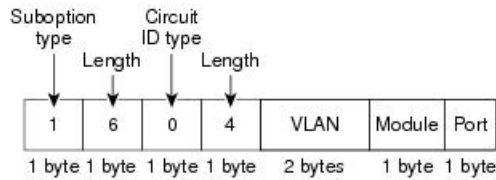
- Circuit-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Circuit-ID type
 - Length of the circuit-ID type
- Remote-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Remote-ID type
 - Length of the remote-ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100/1000 ports and four small form-factor pluggable (SFP) module slots, port 3 is the Gigabit Ethernet 1/0/1 port, port 4 is the Gigabit Ethernet 1/0/2 port, and so forth. Port 27 is the SFP module slot Gigabit Ethernet1/0/25, and so forth.

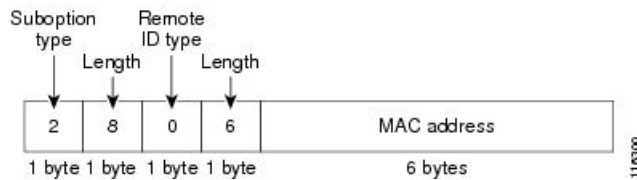
The illustration, *Suboption Packet Formats*, shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. For the circuit-ID suboption, the module number corresponds to the switch number in the stack. The switch uses the packet formats when you globally enable DHCP snooping and enter the `ip dhcp snooping information option global configuration command`.

Figure 17: Suboption Packet Formats

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format



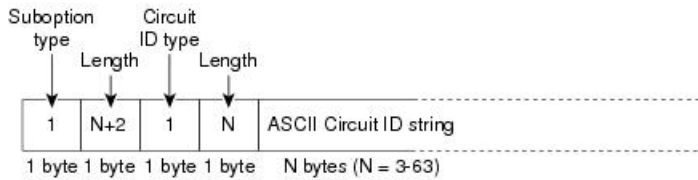
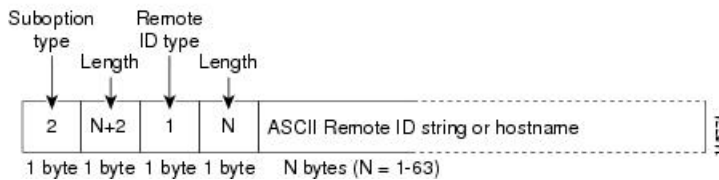
The illustration, *User-Configured Suboption Packet Formats*, shows the packet formats for user-configured remote-ID and circuit-ID suboptions. The switch uses these packet formats when DHCP snooping is globally enabled and when the **ip dhcp snooping information option format remote-id** global configuration command and the **ip dhcp snooping vlan information option format-type circuit-id string** interface configuration command are entered.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

- Circuit-ID suboption fields
 - The circuit-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.

- Remote-ID suboption fields
 - The remote-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.

Figure 18: User-Configured Suboption Packet Formats

Circuit ID Suboption Frame Format (for user-configured string):**Remote ID Suboption Frame Format (for user-configured string):**

Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, address bindings, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 64,000 bindings.

Each database entry (binding) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is

updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and cancel-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
...
<entry-n> <checksum-1-2-...-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The initial-checksum entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

DHCP Snooping and Switch Stacks

DHCP snooping is managed on the active switch. When a new switch joins the stack, the switch receives the DHCP snooping configuration from the active switch. When a member switch leaves the stack, all DHCP snooping address bindings associated with the switch age out.

All snooping statistics are generated on the active switch. If a new active switch is elected, the statistics counters reset.

When a stack merge occurs, all DHCP snooping bindings in the active switch are lost if it is no longer the active switch. With a stack partition, the existing active switch is unchanged, and the bindings belonging to

the partitioned switches age out. The new active switch of the partitioned stack begins processing the new incoming DHCP packets.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 24: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration ⁶
DHCP relay agent	Enabled ⁷
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)
DHCP relay agent forwarding policy	Replace the existing relay agent information
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces ⁸	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration. Note The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration. This feature is operational only when a destination is configured.

⁶ The switch responds to DHCP requests only if it is configured as a DHCP server.

⁷ The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.

- ⁸ Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Configuring the DHCP Server

The switch can act as a DHCP server. If IOS based DHCP server for DHCP clients with management ports are used, both DHCP pool and the corresponding interface must be configured using the Management VRF.

For procedures to configure the switch as a DHCP server, see the “Configuring DHCP” section of the “IP addressing and Services” section of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Server and Switch Stacks

The DHCP binding database is managed on the stack's active switch. When a new active switch is assigned, the new active switch downloads the saved binding database from the TFTP server. When a switchover happens, the new active switch stack will use its database file that has been synced from the old active switch stack using the SSO function. The IP addresses associated with the lost bindings are released. You should configure an automatic backup by using the **ip dhcp database url [timeout seconds | write-delay seconds]** global configuration command.

Configuring the DHCP Relay Agent

Follow these steps to enable the DHCP relay agent on the switch:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	service dhcp Example: Device(config)# <code>service dhcp</code>	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

See the “*Configuring DHCP*” section of the “IP Addressing and Services” section of the *Cisco IOS IP Configuration Guide, Release 12.4* for these procedures:

- Checking (validating) the relay agent information
- Configuring the relay agent forwarding policy

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface vlan <i>vlan-id</i> Example: Device(config)# interface vlan 1	Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 4	ip address <i>ip-address subnet-mask</i> Example: Device(config-if)# ip address 192.108.1.27 255.255.255.0	Configures the interface with an IP address and an IP subnet.
Step 5	ip helper-address <i>address</i> Example: Device(config-if)# ip helper-address 172.16.1.2	Specifies the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests. If you have multiple servers, you can configure one helper address for each server.
Step 6	end Example: Device(config-if)# end	Returns to global configuration mode.
Step 7	Use one of the following: <ul style="list-style-type: none"> • interface range <i>port-range</i> • interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Configures multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configures a single physical port that is connected to the DHCP client, and enter interface configuration mode.

	Command or Action	Purpose
Step 8	switchport mode access Example: <pre>Device(config-if)# switchport mode access</pre>	Defines the VLAN membership mode for the port.
Step 9	switchport access vlan <i>vlan-id</i> Example: <pre>Device(config-if)# switchport access vlan 1</pre>	Assigns the ports to the same VLAN as configured in Step 2.
Step 10	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 11	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 12	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- You must globally enable DHCP snooping on the switch.
- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces. In a service-provider network, a trusted interface is connected to a port on a device in the same network.

- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.
- To use the DHCP snooping option of accepting packets on untrusted inputs, the switch must be an aggregation switch that receives packets with option-82 information from an edge switch.
- The following prerequisites apply to DHCP snooping binding database configuration:
 - You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.
 - Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
 - For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.
 - To ensure that the lease time in the database is accurate, we recommend that you enable and configure Network Time Protocol (NTP).
 - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If you want the switch to relay DHCP packets, the IP address of the DHCP server must be configured on the switch virtual interface (SVI) of the DHCP client.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.

Related Topics

[DHCP Snooping](#)

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the Cisco IOS IP Configuration Guide, Release 12.4

Monitoring DHCP Snooping Information

Table 25: Commands for Displaying DHCP Information

show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
------------------------------	---

show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.



Note If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the statically configured bindings.

Configuring DHCP Server Port-Based Address Allocation

Information About Configuring DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

Default Port-Based Address Allocation Configuration

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

- By default, DHCP server port-based address allocation is disabled.
- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip dhcp snooping database {flash[number]:filename ftp://user:password@host/filename http://[[username:password]@]{hostname / host-ip}[/directory] /image-name.tar rcp://user@host/filename} tftp://host/filename Example: Device(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2	Specifies the URL for the database agent or the binding file by using one of these forms: <ul style="list-style-type: none"> • flash[number]:filename (Optional) Use the <i>number</i> parameter to specify the stack member number of the active switch. The range for <i>number</i> is 1 to 9. • ftp://user:password@host/filename • http://[[username:password]@]{hostname / host-ip}[/directory] /image-name.tar • rcp://user@host/filename • tftp://host/filename
Step 4	ip dhcp snooping database timeout seconds Example: Device(config)# ip dhcp snooping database timeout 300	Specifies (in seconds) how long to wait for the database transfer process to finish before stopping the process.

	Command or Action	Purpose
		The default is 300 seconds. The range is 0 to 86400. Use 0 to define an infinite duration, which means to continue trying the transfer indefinitely.
Step 5	ip dhcp snooping database write-delay <i>seconds</i> Example: <pre>Device(config)# ip dhcp snooping database write-delay 15</pre>	Specifies the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	ip dhcp snooping binding mac-address vlan <i>vlan-id ip-address interface interface-id</i> expiry seconds Example: <pre>Device# ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gi1/1 expiry 1000</pre>	(Optional) Adds binding entries to the DHCP snooping binding database. The <i>vlan-id</i> range is from 1 to 4904. The <i>seconds</i> range is from 1 to 4294967295. Enter this command for each entry that you add. Use this command when you are testing or debugging the switch.
Step 8	show ip dhcp snooping database [detail] Example: <pre>Device# show ip dhcp snooping database detail</pre>	Displays the status and statistics of the DHCP snooping binding database agent.
Step 9	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 10	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling DHCP Server Port-Based Address Allocation

Follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip dhcp use subscriber-id client-id Example: Device(config)# ip dhcp use subscriber-id client-id	Configures the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.
Step 4	ip dhcp subscriber-id interface-name Example: Device(config)# ip dhcp subscriber-id interface-name	Automatically generates a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
Step 5	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 6	ip dhcp server use subscriber-id client-id Example: Device(config-if)# ip dhcp server use subscriber-id client-id	Configures the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

After enabling DHCP port-based address allocation on the switch, use the **ip dhcp pool** global configuration command to preassign IP addresses and to associate them to clients.

Monitoring DHCP Server Port-Based Address Allocation

Table 26: Commands for Displaying DHCP Port-Based Address Allocation Information

Command	Purpose
<code>show interface interface id</code>	Displays the status and configuration of a specific interface.
<code>show ip dhcp pool</code>	Displays the DHCP address pools.
<code>show ip dhcp binding</code>	Displays address bindings on the Cisco IOS DHCP server.



CHAPTER 14

Configuring IP Source Guard

- [Finding Feature Information, on page 251](#)
- [Information About IP Source Guard, on page 251](#)
- [How to Configure IP Source Guard, on page 253](#)
- [Monitoring IP Source Guard, on page 256](#)
- [Additional References, on page 257](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About IP Source Guard

IP Source Guard

You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor and you can enable IP source guard when DHCP snooping is enabled on an untrusted interface.

After IPSPG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping.

The switch uses a source IP lookup table in hardware to bind IP addresses to ports. For IP and MAC filtering, a combination of source IP and source MAC lookups are used. IP traffic with a source IP address in the binding table is allowed, all other traffic is denied.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IPSG is supported only on Layer 2 ports, including access and trunk ports. You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

IP Source Guard for Static Hosts



Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. In a stacked environment, when the active switch failover occurs, the IP source guard entries for static hosts attached to member ports are retained. When you enter the **show device-tracking databaseEXEC** command, the IP device tracking table displays the entries as ACTIVE.



Note Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vendor of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

IP Source Guard Configuration Guidelines

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding mac-address vlan vlan-id ip-address interface interface-id** global configuration command on a routed interface, this error message appears:

Static IP source binding can only be configured on switch port.

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN for that interface.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.



Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- You can enable this feature when 802.1x port-based authentication is enabled.
- When you configure IP source guard smart logging, packets with a source address other than the specified address or an address learned by DHCP are denied, and the packet contents are sent to a NetFlow collector. If you configure this feature, make sure that smart logging is globally enabled.
- In a switch stack, if IP source guard is configured on a stack member interface and you remove the configuration of that switch by entering the **no switch stack-member-number provision** global configuration command, the interface static bindings are removed from the binding table, but they are not removed from the running configuration. If you again provision the switch by entering the **switch stack-member-number provision** command, the binding is restored.

To remove the binding from the running configuration, you must disable IP source guard before entering the **no switch provision** command. The configuration is also removed if the switch reloads while the interface is removed from the binding table.

How to Configure IP Source Guard

Enabling IP Source Guard

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip verify source [mac-check] Example: Device(config-if)# ip verify source	Enables IP source guard with source IP address filtering. (Optional) mac-check —Enables IP Source Guard with source IP address and MAC address filtering.
Step 5	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 6	ip source binding <i>mac-address</i> vlan <i>vlan-id</i> <i>ip-address</i> interface <i>interface-id</i> Example: Device(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1	Adds a static IP source binding. Enter this command for each static binding.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

You must configure the **ip device tracking maximum *limit-number*** interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or by setting an IP device tracking maximum on that interface, IPSG with static hosts rejects all the IP traffic from that interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip device tracking Example: Device(config)# ip device tracking	Turns on the IP host table, and globally enables IP device tracking.
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode.
Step 5	switchport mode access Example: Device(config-if)# switchport mode access	Configures a port as access.
Step 6	switchport access vlan <i>vlan-id</i> Example: Device(config-if)# switchport access vlan 10	Configures the VLAN for this port.

	Command or Action	Purpose
Step 7	ip verify source [tracking] [mac-check] Example: Device(config-if)# ip verify source tracking mac-check	Enables IP source guard with source IP address filtering. (Optional) tracking —Enables IP source guard for static hosts. (Optional) mac-check —Enables MAC address filtering. The command ip verify source tracking mac-check enables IP source guard for static hosts with MAC address filtering.
Step 8	ip device tracking maximum <i>number</i> Example: Device(config-if)# ip device tracking maximum 8	Establishes a maximum limit for the number of static IPs that the IP device tracking table allows on the port. The range is 1 to 10. The maximum number is 10. Note You must configure the ip device tracking maximum limit-number interface configuration command.
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode.

Monitoring IP Source Guard

Table 27: Privileged EXEC show Commands

Command	Purpose
show ip verify source [interface <i>interface-id</i>]	Displays the IP source guard configuration on the switch or on a specific interface.
show ip device tracking { all interface <i>interface-id</i> ip <i>ip-address</i> mac <i>mac-address</i> }	Displays information about the entries in the IP device tracking table.

Table 28: Interface Configuration Commands

Command	Purpose
ip verify source tracking	Verifies the data source.

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 15

Configuring Dynamic ARP Inspection

- Finding Feature Information, on page 259
- Restrictions for Dynamic ARP Inspection, on page 259
- Understanding Dynamic ARP Inspection, on page 261
- Default Dynamic ARP Inspection Configuration, on page 264
- Relative Priority of ARP ACLs and DHCP Snooping Entries, on page 265
- Configuring ARP ACLs for Non-DHCP Environments , on page 265
- Configuring Dynamic ARP Inspection in DHCP Environments, on page 267
- Limiting the Rate of Incoming ARP Packets, on page 270
- Performing Dynamic ARP Inspection Validation Checks, on page 271
- Monitoring DAI, on page 273
- Verifying the DAI Configuration, on page 273
- Additional References, on page 274

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.

- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.

- Dynamic ARP inspection is supported on access ports, trunk ports, and EtherChannel ports.



Note Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection packets might not reach the RSPAN destination port.

- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.

- The rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.

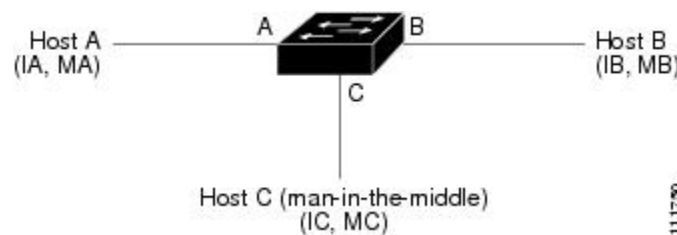
- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.
- When you enable dynamic ARP inspection on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.

Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 26-1 shows an example of ARP cache poisoning.

Figure 19: ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the-middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection 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

Dynamic ARP inspection 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.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan** *vlan-range* global configuration command.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command.

You can configure dynamic ARP inspection 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. Use the **ip arp inspection validate** **{[src-mac] [dst-mac] [ip]}** global configuration command.

Interface Trust States and Network Security

Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the **ip arp inspection trust interface** configuration command.

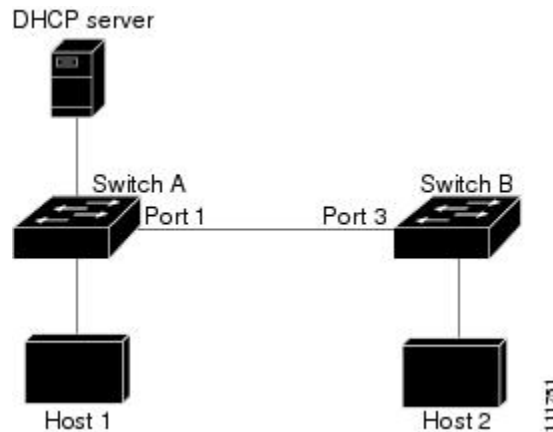


Caution

Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In the following figure, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 20: ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches.



Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.



Note The rate limit for an EtherChannel is applied separately to each switch in a stack. For example, if a limit of 20 pps is configured on the EtherChannel, each switch with ports in the EtherChannel can carry up to 20 pps. If any switch exceeds the limit, the entire EtherChannel is placed into the error-disabled state.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command.

Default Dynamic ARP Inspection Configuration

Feature	Default Settings
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.

Feature	Default Settings
Log buffer	<p>When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged.</p> <p>The number of entries in the log is 32.</p> <p>The number of system messages is limited to 5 per second.</p> <p>The logging-rate interval is 1 second.</p>
Per-VLAN logging	All denied or dropped ARP packets are logged.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the `ip arp inspection filter vlan` global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 2 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

Procedure

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p>	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<code>arp access-list <i>acl-name</i></code>	<p>Defines an ARP ACL, and enters ARP access-list configuration mode. By default, no ARP access lists are defined.</p> <p>Note At the end of the ARP access list, there is an implicit deny ip any mac any command.</p>
Step 4	<code>permit ip host <i>sender-ip</i> mac host <i>sender-mac</i></code>	<p>Permits ARP packets from the specified host (Host 2).</p> <ul style="list-style-type: none"> • For <i>sender-ip</i>, enter the IP address of Host 2. • For <i>sender-mac</i>, enter the MAC address of Host 2.
Step 5	<code>exit</code>	Returns to global configuration mode.
Step 6	<code>ip arp inspection filter <i>arp-acl-name</i> vlan <i>vlan-range</i> [static]</code>	<p>Applies ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN.</p> <ul style="list-style-type: none"> • For <i>arp-acl-name</i>, specify the name of the ACL created in Step 2. • For <i>vlan-range</i>, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify static to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. <p>If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL.</p> <p>ARP packets containing only IP-to-MAC address bindings are compared against the</p>

	Command or Action	Purpose
		ACL. Packets are permitted only if the access list permits them.
Step 7	interface <i>interface-id</i>	Specifies Switch A interface that is connected to Switch B, and enters the interface configuration mode.
Step 8	no ip arp inspection trust	Configures Switch A interface that is connected to Switch B as untrusted. By default, all interfaces are untrusted. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the ip arp inspection vlan logging global configuration command.
Step 9	end	Returns to privileged EXEC mode.
Step 10	Use the following show commands: <ul style="list-style-type: none"> • show arp access-list <i>acl-name</i> • show ip arp inspection vlan <i>vlan-range</i> • show ip arp inspection interfaces 	Verifies your entries.
Step 11	show running-config Example: Device# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Dynamic ARP Inspection in DHCP Environments

Before you begin

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B. Both switches are running dynamic

ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.



Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

Follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show cdp neighbors Example: Device (config-if) # show cdp neighbors	Verify the connection between the switches.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	ip arp inspection vlan <i>vlan-range</i> Example: Device (config) # ip arp inspection vlan 1	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For <i>vlan-range</i> , specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 5	Interface <i>interface-id</i> Example: Device (config) # interface gigabitethernet1/0/1	Specifies the interface connected to the other switch, and enter interface configuration mode.
Step 6	ip arp inspection trust Example: Device (config-if) # ip arp inspection trust	Configures the connection between the switches as trusted. By default, all interfaces are untrusted.

	Command or Action	Purpose
		<p>The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets.</p> <p>For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the <code>ip arp inspection vlan logging global</code> configuration command.</p>
Step 7	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 8	show ip arp inspection interfaces Example:	Verifies the dynamic ARP inspection configuration on interfaces.
Step 9	show ip arp inspection vlan <i>vlan-range</i> Example: Device (config-if) # show ip arp inspection vlan 1	Verifies the dynamic ARP inspection configuration on VLAN.
Step 10	show ip dhcp snooping binding Example: Device (config-if) # show ip dhcp snooping binding	Verifies the DHCP bindings.
Step 11	show ip arp inspection statistics vlan <i>vlan-range</i> Example: Device (config-if) # show ip arp inspection statistics vlan 1	Checks the dynamic ARP inspection statistics on VLAN.
Step 12	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 13	configure terminal Example: Device# configure terminal	Enters global configuration mode.

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.



Note

Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

Follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Specifies the interface to be rate-limited, and enter interface configuration mode.
Step 4	ip arp inspection limit {rate pps [burst interval seconds] none}	Limits the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted interfaces and unlimited on trusted interfaces. The burst interval is 1 second. <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • For ratepps, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. • (Optional) For burst intervalseconds, specify the consecutive interval in

	Command or Action	Purpose
		seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. • For rate none , specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	exit	Returns to global configuration mode.
Step 6	Use the following commands: • errdisable detect cause arp-inspection • errdisable recovery cause arp-inspection • errdisable recovery interval interval	(Optional) Enables error recovery from the dynamic ARP inspection error-disabled state, and configure the dynamic ARP inspection recover mechanism variables. By default, recovery is disabled, and the recovery interval is 300 seconds. For interval interval , specify the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 7	exit	Returns to privileged EXEC mode.
Step 8	Use the following show commands: • show ip arp inspection interfaces • show errdisable recovery	Verifies your settings.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Performing Dynamic ARP Inspection Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip arp inspection validate {[src-mac] [dst-mac] [ip]}	Performs a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: <ul style="list-style-type: none"> • For src-mac, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For dst-mac, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For ip, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables src and dst mac validations, and a second command enables IP validation only, the src and dst mac validations are disabled as a result of the second command.
Step 4	exit	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	<code>show ip arp inspection vlan <i>vlan-range</i></code>	Verifies your settings.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Description
<code>clear ip arp inspection statistics</code>	Clears dynamic ARP inspection statistics.
<code>show ip arp inspection statistics [vlan <i>vlan-range</i>]</code>	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
<code>clear ip arp inspection log</code>	Clears the dynamic ARP inspection log buffer.
<code>show ip arp inspection log</code>	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the `show ip arp inspection statistics` command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Description
<code>show arp access-list [acl-name]</code>	Displays detailed information about ARP ACLs.

Command	Description
<code>show ip arp inspection interfaces [interface-id]</code>	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
<code>show ip arp inspection vlan <i>vlan-range</i></code>	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 16

Configuring IPv6 First Hop Security

- Finding Feature Information, on page 275
- Prerequisites for First Hop Security in IPv6, on page 275
- Restrictions for First Hop Security in IPv6, on page 276
- Information about First Hop Security in IPv6, on page 276
- How to Configure an IPv6 Snooping Policy, on page 278
- How to Attach an IPv6 Snooping Policy to an Interface, on page 280
- How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface, on page 282
- How to Attach an IPv6 Snooping Policy to VLANs Globally , on page 283
- **How to Configure the IPv6 Binding Table Content** , on page 283
- How to Configure an IPv6 Neighbor Discovery Inspection Policy, on page 284
- How to Configure an IPv6 Router Advertisement Guard Policy, on page 288
- **How to Configure an IPv6 DHCP Guard Policy** , on page 293
- How to Configure IPv6 Source Guard, on page 298
- How to Configure IPv6 Prefix Guard, on page 301
- Configuration Examples for IPv6 First Hop Security, on page 303

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.
- You should be familiar with the IPv6 neighbor discovery feature.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):
 - A physical port with an FHS policy attached cannot join an EtherChannel group.
 - An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:
 - Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
 - Configure a snooping policy with a lower security-level, for example glean or inspect. However; configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, an EtherChannel interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.



Note Effective Cisco IOS XE Release 16.3.1, ND Inspection functionality, IPv6 Snooping Policy, and IPv6 FHS Binding Table Content are supported through Switch Integrated Security Feature (SISF)-based Device Tracking. For more information, see *Configuring SISF based device tracking* section of the Software Configuration Guide.

- **IPv6 Router Advertisement Guard**—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.
- **IPv6 DHCP Guard**—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.
- **IPv6 Source Guard**—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

The IPv6 source guard feature provides the ability to store entries in the hardware TCAM table to prevent a host from sending packets with an invalid IPv6 source address.

To debug source-guard packets, use the `debug ipv6 snooping source-guard` privileged EXEC command.



Note The IPv6 source guard and prefix guard features are supported only in the ingress direction; it is not supported in the egress direction.

The following restrictions apply:

- An FHS policy cannot be attached to an physical port when it is a member of an EtherChannel group.
- When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
- An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
- When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use **ip verify source mac-check** instead of **ip verify source**. IPv4 connectivity on a given port might

break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).

- You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
- PVLAN and Source/Prefix Guard cannot be applied together.
- IPv6 Source Guard and Prefix Guard is supported on EtherChannels

For more information on IPv6 Source Guard, see the [IPv6 Source Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

For more information on IPv6 Prefix Guard, see the [IPv6 Prefix Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.



Note IPv6 Destination Guard is recommended to apply on Layer 2 VLAN with an SVI configured

For more information about IPv6 Destination Guard, see the [IPv6 Destination Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

How to Configure an IPv6 Snooping Policy

The IPv6 Snooping Policy feature is deprecated starting from Cisco IOS XE Denali 16.3.1. Although the commands are visible on the CLI and you can configure them, we recommend that you use the Switch Integrated Security Feature (SISF)-based Device Tracking feature instead.

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	ipv6 snooping policy <i>policy-name</i> Example: Device(config)# ipv6 snooping policy example_policy	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
Step 3	<pre>{[default] [device-role {node switch}] [limit address-count <i>value</i>] [no] [protocol {dhcp ndp}] [security-level {glean guard inspect}] [tracking {disable [stale-lifetime [<i>seconds</i> infinite] enable [reachable-lifetime [<i>seconds</i> infinite]}]}] [trusted-port] }</pre> Example: Device (config-ipv6-snooping) # security-level inspect Example: Device (config-ipv6-snooping) # trusted-port	Enables data address gleaning, validates messages against various criteria, specifies the security level for messages. <ul style="list-style-type: none"> • (Optional) default—Sets all to default options. • (Optional) device-role{node switch}—Specifies the role of the device attached to the port. Default is node. • (Optional) limit address-count <i>value</i>—Limits the number of addresses allowed per target. • (Optional) no—Negates a command or sets it to defaults. • (Optional) protocol{dhcp ndp}—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is dhcp and ndp. To change the default, use the no protocol command. • (Optional) security-level{glean guard inspect}—Specifies the level of security enforced by the feature. Default is guard. <ul style="list-style-type: none"> glean—Gleans addresses from messages and populates the binding table without any verification. guard—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. inspect—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) tracking {disable enable}—Overrides the default tracking behavior and specifies a tracking option.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) trusted-port—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	end Example: Device(config-ipv6-snooping)# exit	Exits configuration modes to Privileged EXEC mode.
Step 5	show ipv6 snooping policy <i>policy-name</i> Example: Device# show ipv6 snooping policy example_policy	Displays the snooping policy configuration.

What to do next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	switchport Example:	Enters the Switchport mode.

	Command or Action	Purpose
	<pre>Device(config-if)# switchport</pre>	<p>Note To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the switchport interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration. The command prompt displays as (config-if)# in Switchport configuration mode.</p>
<p>Step 4</p>	<pre>ipv6 snooping [attach-policy policy_name [vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids}] vlan {vlan_id add vlan_ids exceptvlan_ids none remove vlan_ids all}]</pre> <p>Example:</p> <pre>Device(config-if)# ipv6 snooping</pre> <p>or</p> <pre>Device(config-if)# ipv6 snooping attach-policy example_policy</pre> <p>or</p> <pre>Device(config-if)# ipv6 snooping vlan 111,112</pre> <p>or</p> <pre>Device(config-if)# ipv6 snooping attach-policy example_policy vlan 111,112</pre>	<p>Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the ipv6 snooping command without the attach-policy keyword. To attach the default policy to VLANs on the interface, use the ipv6 snooping vlan command. The default policy is, security-level guard, device-role node, protocol ndp and dhcp.</p>
<p>Step 5</p>	<pre>do show running-config</pre> <p>Example:</p> <pre>Device#(config-if)# do show running-config</pre>	<p>Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.</p>

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface range Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 snooping [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if-range)# ipv6 snooping attach-policy example_policy or Device(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 snooping vlan 222, 223,224	Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Snooping Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping Policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Device(config)# <code>vlan configuration 333</code>	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 snooping [attach-policy <i>policy_name</i>] Example: Device(config-vlan-config)# <code>ipv6 snooping attach-policy example_policy</code>	Attaches the IPv6 Snooping policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, security-level guard , device-role node , protocol ndp and dhcp .
Step 4	do show running-config Example: Device#(config-if)# <code>do show running-config</code>	Verifies that the policy is attached to the specified VLANs without exiting the interface configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 2	[no] ipv6 neighbor binding [vlan <i>vlan-id</i> { <i>ipv6-address</i> interface <i>interface_type</i>	Adds a static entry to the binding table database.

	Command or Action	Purpose
	<code>stack/module/port hw_address</code> <code>[reachable-lifetimevalue [seconds default infinite] [tracking { [default disable] [reachable-lifetimevalue [seconds default infinite] [enable [reachable-lifetimevalue [seconds default infinite] [retry-interval {seconds default [reachable-lifetimevalue [seconds default infinite] } }]</code> Example: <code>Device(config)# ipv6 neighbor binding</code>	Note Switch adds small variance to configured reachable-time value to improve system stability during timer expiry of binding entries.
Step 3	<code>[no] ipv6 neighbor binding max-entries number [mac-limit number port-limit number [mac-limit number] vlan-limit number [mac-limit number] [port-limit number [mac-limit number]]]]</code> Example: <code>Device(config)# ipv6 neighbor binding max-entries 30000</code>	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	<code>ipv6 neighbor binding logging</code> Example: <code>Device(config)# ipv6 neighbor binding logging</code>	Enables the logging of binding table main events.
Step 5	<code>exit</code> Example: <code>Device(config)# exit</code>	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	<code>show ipv6 neighbor binding</code> Example: <code>Device# show ipv6 neighbor binding</code>	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code> Example: <code>Device# configure terminal</code>	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	[no]ipv6 nd inspection policy <i>policy-name</i> Example: Device(config)# ipv6 nd inspection policy example_policy	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	device-role {host switch} Example: Device(config-nd-inspection)# device-role switch	Specifies the role of the device attached to the port. The default is host .
Step 4	limit address-count <i>value</i> Example: Device(config-nd-inspection)# limit address-count 1000	Enter 1–10,000.
Step 5	tracking {enable [reachable-lifetime { <i>value</i> infinite}] disable [stale-lifetime { <i>value</i> infinite}]} Example: Device(config-nd-inspection)# tracking disable stale-lifetime infinite	Overrides the default tracking policy on a port.
Step 6	trusted-port Example: Device(config-nd-inspection)# trusted-port	Configures a port to become a trusted port.
Step 7	validate source-mac Example: Device(config-nd-inspection)# validate source-mac	Checks the source media access control (MAC) address against the link-layer address.
Step 8	no {device-role limit address-count tracking trusted-port validate source-mac} Example: Device(config-nd-inspection)# no validate source-mac	Remove the current configuration of a parameter with the no form of the command.
Step 9	default {device-role limit address-count tracking trusted-port validate source-mac} Example: Device(config-nd-inspection)# default limit address-count	Restores configuration to the default values.

	Command or Action	Purpose
Step 10	do show ipv6 nd inspection policy <i>policy_name</i> Example: Device(config-nd-inspection)# do show ipv6 nd inspection policy example_policy	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}] Example: Device(config-if)# ipv6 nd inspection attach-policy example_policy or Device(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224 or Device(config-if)# ipv6 nd inspection vlan 222, 223,224	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
Step 4	do show running-config Example: Device#(config-if)# do show running-config	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd inspection [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}] Example: Device(config-if-range)# ipv6 nd inspection attach-policy example_policy or Device(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 nd inspection vlan 222, 223,224	Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Device(config)# vlan configuration 334	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 nd inspection [attach-policy <i>policy_name</i>] Example: Device(config-vlan-config)# ipv6 nd inspection attach-policy example_policy	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role host , no drop-unsecure, limit address-count disabled, sec-level minimum is disabled, tracking is disabled, no trusted-port, no validate source-mac.
Step 4	do show running-config Example: Device#(config-if)# do show running-config	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 nd rguard policy <i>policy-name</i> Example: Device(config)# ipv6 nd rguard policy example_policy	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[no]device-role {host monitor router switch} Example: Device(config-nd-rguard) # device-role switch	Specifies the role of the device attached to the port. The default is host . Note For a network with both host-facing ports and router-facing ports, along with a RA guard policy configured with device-role host on host-facing ports or vlan, it is mandatory to configure a RA guard policy with device-role router on router-facing ports to allow the RA Guard feature to work properly.
Step 4	[no]hop-limit {maximum minimum} <i>value</i> Example: Device(config-nd-rguard) # hop-limit maximum 33	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host, prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure minimum to block RA messages with Hop Limit values lower than the value you specify. Configure maximum to block RA messages with Hop Limit values greater than the value you specify.
Step 5	[no]managed-config-flag {off on} Example: Device(config-nd-rguard) # managed-config-flag on	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rogue RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.

	Command or Action	Purpose
		<p>On—Accepts and forwards RA messages with an M value of 1, blocks those with 0.</p> <p>Off—Accepts and forwards RA messages with an M value of 0, blocks those with 1.</p>
Step 6	<p>[no]match {ipv6 access-list list ra prefix-list list}</p> <p>Example:</p> <pre>Device(config-nd-raguard)# match ipv6 access-list example_list</pre>	Matches a specified prefix list or access list.
Step 7	<p>[no]other-config-flag {on off}</p> <p>Example:</p> <pre>Device(config-nd-raguard)# other-config-flag on</pre>	<p>Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rouge RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.</p> <p>On—Accepts and forwards RA messages with an O value of 1, blocks those with 0.</p> <p>Off—Accepts and forwards RA messages with an O value of 0, blocks those with 1.</p>
Step 8	<p>[no]router-preference maximum {high medium low}</p> <p>Example:</p> <pre>Device(config-nd-raguard)# router-preference maximum high</pre>	<p>Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled.</p> <ul style="list-style-type: none"> • high—Accepts RA messages with the Router Preference set to high, medium, or low. • medium—Blocks RA messages with the Router Preference set to high. • low—Blocks RA messages with the Router Preference set to medium and high.
Step 9	<p>[no]trusted-port</p> <p>Example:</p> <pre>Device(config-nd-raguard)# trusted-port</pre>	When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.
Step 10	<p>default {device-role hop-limit {maximum minimum} managed-config-flag match {ipv6 access-list ra prefix-list } other-config-flag router-preference maximum trusted-port}</p> <p>Example:</p> <pre>Device(config-nd-raguard)# default hop-limit</pre>	Restores a command to its default value.

	Command or Action	Purpose
Step 11	do show ipv6 nd rguard policy <i>policy_name</i> Example: Device(config-nd-raguard)# do show ipv6 nd rguard policy example_policy	(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 nd rguard [attach-policy <i>policy_name</i> [vlan {<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}] vlan [{<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}] Example: Device(config-if)# ipv6 nd rguard attach-policy example_policy or Device(config-if)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 or Device(config-if)# ipv6 nd rguard vlan 222, 223,224	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Device#(config-if)# do show running-config	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd rguard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if-range)# ipv6 nd rguard attach-policy example_policy or Device(config-if-range)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 nd rguard vlan 222, 223,224	Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to VLANs regardless of interface:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Device(config)# <code>vlan configuration 335</code>	Specifies the VLANs to which the IPv6 RA Guard policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy <i>policy_name</i>] Example: Device(config-vlan-config)# <code>ipv6 nd raguard attach-policy example_policy</code>	Attaches the IPv6 RA Guard policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Device#(config-if)# <code>do show running-config</code>	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 2	[no]ipv6 dhcp guard policy <i>policy-name</i> Example: Device(config)# <code>ipv6 dhcp guard policy example_policy</code>	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.

	Command or Action	Purpose
Step 3	<p>[no]device-role {client server}</p> <p>Example:</p> <pre>Device(config-dhcp-guard)# device-role server</pre>	<p>(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is client.</p> <ul style="list-style-type: none"> • client—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • server—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.
Step 4	<p>[no] match server access-list <i>ipv6-access-list-name</i></p> <p>Example:</p> <pre>;;Assume a preconfigured IPv6 Access List as follows: Device(config)# ipv6 access-list my_acls Device(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any ;;configure DCHPv6 Guard to match approved access list. Device(config-dhcp-guard)# match server access-list my_acls</pre>	<p>(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.</p>
Step 5	<p>[no] match reply prefix-list <i>ipv6-prefix-list-name</i></p> <p>Example:</p> <pre>;;Assume a preconfigured IPv6 prefix list as follows: Device(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128 ;; Configure DCHPv6 Guard to match prefix Device(config-dhcp-guard)# match reply prefix-list my_prefix</pre>	<p>(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.</p>
Step 6	<p>[no]preference{ max limit min limit }</p> <p>Example:</p> <pre>Device(config-dhcp-guard)# preference max 250 Device(config-dhcp-guard)#preference min 150</pre>	<p>Configure max and min when device-role is server to filter DCHPv6 server advertisements by the server preference value. The defaults permit all advertisements.</p> <p>max limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed.</p>

	Command or Action	Purpose
		min limit —(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.
Step 7	[no] trusted-port Example: Device(config-dhcp-guard)# trusted-port	(Optional) trusted-port —Sets the port to a trusted mode. No further policing takes place on the port. Note If you configure a trusted port then the device-role option is not available.
Step 8	default {device-role trusted-port} Example: Device(config-dhcp-guard)# default device-role	(Optional) default —Sets a command to its defaults.
Step 9	do show ipv6 dhcp guard policy policy_name Example: Device(config-dhcp-guard)# do show ipv6 dhcp guard policy example_policy	(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submenu. Omitting the <i>policy_name</i> variable displays all DHCPv6 policies.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy poll
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy poll vlan add 1
 vlan 1
 ipv6 dhcp guard attach-policy poll
show ipv6 dhcp guard policy poll
```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if)# ipv6 dhcp guard attach-policy example_policy or Device(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or Device(config-if)# ipv6 dhcp guard vlan 222, 223,224	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface Interface_type <i>stack/module/port</i> Example: Device#(config-if)# do show running-config gig 1/1/4	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 dhcp guard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy or Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 dhcp guard vlan 222, 223,224	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to VLANs Globally

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 2	vlan configuration <i>vlan_list</i> Example: Device(config)# <code>vlan configuration 334</code>	Specifies the VLANs to which the IPv6 Snooping policy will be attached ; enters the VLAN interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy <i>policy_name</i>] Example: Device(config-vlan-config)# <code>ipv6 dhcp guard attach-policy example_policy</code>	Attaches the IPv6 Neighbor Discovery policy to the specified VLANs across all switch and stack interfaces. The default policy is attached if the attach-policy option is not used. The default policy is, device-role client , no trusted-port.
Step 4	do show running-config Example: Device#(config-if)# <code>do show running-config</code>	Confirms that the policy is attached to the specified VLANs without exiting the configuration mode.

How to Configure IPv6 Source Guard

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 3	[no] ipv6 source-guard policy <i>policy_name</i> Example: Device(config)# <code>ipv6 source-guard policy example_policy</code>	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}]	(Optional) Defines the IPv6 Source Guard policy.

	Command or Action	Purpose
	Example: Device(config-sisf-sourceguard)# deny global-autoconf	<ul style="list-style-type: none"> • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the administrator wants to block hosts with self-configured addresses to send traffic. • permit link-local—Allows all data traffic that is sourced by a link-local address. <p>Note Trusted option under source guard policy is not supported.</p>
Step 5	end Example: Device(config-sisf-sourceguard)# end	Exits out of IPv6 Source Guard policy configuration mode.
Step 6	show ipv6 source-guard policy <i>policy_name</i> Example: Device# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

What to do next

Apply the IPv6 Source Guard policy to an interface.

How to Attach an IPv6 Source Guard Policy to an Interface

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.

	Command or Action	Purpose
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name Example: Device#(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface port-channel port-channel-number Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if) # ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name Example: Device(config-if) # show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to Configure IPv6 Prefix Guard



Note To allow routing protocol control packets sourced by a link-local address when prefix guard is applied, enable the permit link-local command in the source-guard policy configuration mode.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	[no] ipv6 source-guard policy <i>source-guard-policy</i> Example: Device(config)# ipv6 source-guard policy my_snooping_policy	Defines an IPv6 source-guard policy name and enters switch integrated security features source-guard policy configuration mode.
Step 4	[no] validate address Example: Device(config-sisf-sourceguard)# no validate address	Disables the validate address feature and enables the IPv6 prefix guard feature to be configured.
Step 5	validate prefix Example: Device(config-sisf-sourceguard)# validate prefix	Enables IPv6 source guard to perform the IPv6 prefix-guard operation.
Step 6	exit Example: Device(config-sisf-sourceguard)# exit	Exits switch integrated security features source-guard policy configuration mode and returns to privileged EXEC mode.
Step 7	show ipv6 source-guard policy <i>[source-guard-policy]</i> Example:	Displays the IPv6 source-guard policy configuration.

	Command or Action	Purpose
	Device# <code>show ipv6 source-guard policy policy1</code>	

How to Attach an IPv6 Prefix Guard Policy to an Interface

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 3	interface <i>Interface_type stack/module/port</i> Example: Device(config)# <code>interface gigabitethernet 1/1/4</code>	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard attach-policy <i>policy_name</i> Example: Device(config-if)# <code>ipv6 source-guard attach-policy example_policy</code>	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if)# <code>show ipv6 source-guard policy example_policy</code>	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface port-channel <i>port-channel-number</i> Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

Configuration Examples for IPv6 First Hop Security

Examples: How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch(config-sisf-sourceguard) # validate address
switch(config-sisf-sourceguard)# exit
Switch(config)# interface Po4
Switch(config)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
Switch(config-if)# exit
switch(config)#
```

Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch (config-sisf-sourceguard)# no validate address
Switch((config-sisf-sourceguard)# validate prefix
Switch(config)# interface Po4
```

Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

```
Switch(config-if)# ipv6 snooping  
Switch(config-if)# ipv6 source-guard attach-policy POL
```



CHAPTER 17

Configuring SISF-Based Device Tracking

- [Information About SISF-Based Device Tracking, on page 305](#)
- [How to Configure SISF-Based Device Tracking, on page 311](#)
- [Configuration Examples for SISF-Based Device Tracking, on page 319](#)
- [Feature History and Information for SISF-Based Device Tracking, on page 322](#)

Information About SISF-Based Device Tracking

Overview of SISF-Based Device Tracking

The Switch Integrated Security Features based (SISF-based) device tracking feature is part of the suite of first-hop security features.

The main role of the feature is to track the presence, location, and movement of end-nodes in the network. SISF snoops traffic received by the switch, extracts device identity (MAC and IP address), and stores them in a binding table. Many features, such as, IEEE 802.1X, web authentication, Cisco TrustSec and LISP etc., depend on the accuracy of this information to operate properly.

SISF-based device tracking supports both IPv4 and IPv6.

Even with the introduction of SISF-based device tracking, the legacy device tracking CLI (IP Device Tracking (IPDT) and IPv6 Snooping CLI) continues to be available. When you bootup the switch, the set of commands that is available depends on existing configuration, and only one of the following is available:

- SISF-based device tracking CLI, or
- IPDT and IPv6 Snooping CLI



Note The IPDT and IPv6 Snooping commands are deprecated, but continue to be available. We recommend that you upgrade to SISF-based device tracking.

If you are using the IPDT and IPv6 Snooping CLI and want to migrate to SISF-based device tracking, see *Migrating from legacy IPDT and IPv6 Snooping to SISF-Based Device Tracking*, for more information.

SISF-based device tracking can be enabled manually (by using **device-tracking** commands), or programmatically (which is the case when providing device tracking services to other features).

Options to Enable SISF-Based Device Tracking

SISF-Based device tracking is disabled by default.

You can enable it by defining a device tracking policy and attaching the policy to a specific target.



Note The target could be an interface or a VLAN.

Manually Enabling SISF-Based Device Tracking

- Option 1: Apply the **default** device tracking policy to a target.

Enter the **device-tracking** command in the interface configuration mode or in the VLAN configuration mode. The system then attaches the **default** policy it to the interface or VLAN.



Note The **default** policy is a built-in policy with default settings; you cannot change any of the attributes of the **default** policy. In order to be able to configure device tracking policy attributes you must create a custom policy. See *Option 2: Create a custom policy with custom settings*.

- Option 2: Create a custom policy with custom settings.

Enter the device-tracking policy command in global configuration mode and enter a custom policy name. The system creates a policy with the name you specify. You can then configure the available settings, in the device tracking configuration mode (config-device-tracking), and attach the policy to a specified target.

Programmatically Enabling SISF-Based Device Tracking

Some features rely on device tracking and utilize the trusted database of binding entries that SISF-based device tracking builds and maintains. These features, also called device tracking clients, enable device tracking programmatically (create and attach the device tracking policy).



Note The exceptions here are IEEE 802.1X, web authentication, Cisco TrustSec, and IP Source Guard (IPSG) - they also rely on device tracking, but they do not enable it. For these device tracking clients, you must enter the **ip dhcp snooping vlan *vlan*** command, to programmatically enable device tracking on a particular target.

Note the following about programmatically enabling SISF-based device tracking:

- A device tracking client *requires* device tracking to be enabled.

There are several device tracking clients, therefore, multiple programmatic policies could be created. The settings of each policy differ depending on the device tracking client that creates the policy.

- The policy that is created, and its settings, are system-defined.

Configurable policy attributes are available in the device tracking configuration mode (config-device-tracking) and vary from one release to another. If you try to modify an attribute that is not configurable, the configuration change is rejected and an error message is displayed.

For release-specific information about programmatically created policies, see *Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE <release name> <release number>* in the required version of the document.

Migrating from Legacy Commands to SISF-Based Device-Tracking Commands

Migrating from Legacy IPDT and IPv6 Snooping to SISF-Based Device Tracking

Starting with Cisco IOS XE Denali 16.1.1, the existing IPv6 snooping and IP Device Tracking (IPDT) commands have corresponding SISF-based device-tracking commands that allow you to apply your configuration to both IPv4 and IPv6 address families.

After you have upgraded from a Cisco IOS XE 3.x.x release to a Cisco IOS XE 16.x.x release, enter the **device-tracking upgrade-cli** to convert legacy IPDT and IPv6 Snooping commands to SISF-based device tracking commands. After you run the command, only the new device-tracking commands are available on your device and the legacy commands are not supported.

Based on the legacy configuration that exists on your device, the **device-tracking upgrade-cli** command upgrades your CLI differently. Consider the following configuration scenarios and the corresponding migration results before you migrate your existing configuration.



Note You cannot configure a mix of the old IPDT and IPv6 snooping CLI with the new SISF-based device-tracking CLI.

Only IPDT Configuration Exists

If your device has only IPDT configuration, running the **device-tracking upgrade-cli** command converts the configuration to use the new SISF policy that is created and attached to the interface. You can then update this SISF policy.

If you continue to use the legacy commands, this restricts you to operate in a legacy mode where only the legacy IPDT and IPv6 snooping commands are available on the device.

Only IPv6 Snooping Configuration Exists

On a device with existing IPv6 snooping configuration, the old IPv6 Snooping commands are available for further configuration. The following options are available:

- (Recommended) Use the **device-tracking upgrade-cli** command to convert all your legacy configuration to the new SISF-based device tracking commands. After conversion, only the new device tracking commands will work on your device.
- Use the legacy IPv6 Snooping commands for your future configuration and do not run the **device-tracking upgrade-cli** command. With this option, only the legacy IPv6 Snooping commands are available on your device, and you cannot use the new SISF-based device tracking CLI commands.

Both IPDT and IPv6 Snooping Configuration Exist

On a device that has both legacy IPDT configuration and IPv6 snooping configuration, you can convert legacy commands to the SISF-based device tracking CLI commands. However, note that only one snooping policy can be attached to an interface, and the IPv6 snooping policy parameters override the IPDT settings.



Note If you do not migrate to the new SISF-based commands and continue to use the legacy IPv6 snooping or IPDT commands, your IPv4 device tracking configuration information may be displayed in the IPv6 snooping commands, as the SISF-based device tracking feature handles both IPv4 and IPv6 configuration. To avoid this, we recommend that you convert your legacy configuration to SISF-based device tracking commands.

No IPDT or IPv6 Snooping Configuration Exists

If your device has no legacy IP Device Tracking or IPv6 Snooping configurations, you can use only the new SISF-based device tracking commands for all your future configuration. The legacy IPDT commands and IPv6 snooping commands are not available.



Note Starting from Cisco IOS XE Denali 16.3.1, the **ip dhcp snooping vlan** *vlan* command creates a device tracking policy programmatically, to support the IEEE 802.1X, web authentication, Cisco TrustSec and IPSG features. The programmatically created policy tracks both IPv4 and IPv6 clients. Ensure that this command is configured, if you are using any of the aforementioned features.

IPDT, IPv6 Snooping, and SISF-Based Device Tracking CLI Compatibility

Table [Table 29: IPDT → IPv6 Snooping Commands, on page 308](#) displays legacy IPDT and the IPv6 snooping commands they are converted to. (The commands listed here are applicable if you have not upgraded to SISF-based device-tracking).

Table [Table 30: IPDT → SISF Commands, on page 309](#) displays legacy IPDT and the SISF-based device-tracking commands. (The commands listed here are applicable if you have upgraded to SISF-based device-tracking, with the **device-tracking upgrade-cli** command.)

Table 29: IPDT → IPv6 Snooping Commands

Legacy IP Device Tracking (IPDT)	IPv6 Snooping Command (Until Cisco IOS XE Denali 16.3.6 and Cisco IOS XE Everest 16.5.x)	IPv6 Snooping Command (Starting from Cisco IOS XE Denali 16.3.7 and all later releases except Cisco IOS XE Everest 16.5.x).
ip device tracking probe count	Set to the default value, and cannot be changed.	Set to the default value, and cannot be changed.
ip device tracking probe delay	ipv6 neighbor binding reachable-lifetime Attention Incorrect system conversion. ²	Set to the default value, and cannot be changed

Legacy IP Device Tracking (IPDT)	IPv6 Snooping Command (Until Cisco IOS XE Denali 16.3.6 and Cisco IOS XE Everest 16.5.x)	IPv6 Snooping Command (Starting from Cisco IOS XE Denali 16.3.7 and all later releases except Cisco IOS XE Everest 16.5.x).
ip device tracking probe interval	ipv6 snooping tracking retry-interval Attention Incorrect system conversion. 10	ipv6 neighbor binding reachable-lifetime
ip device tracking probe use-svi	Set to the default behavior, and cannot be changed.	Set to the default behavior, and cannot be changed.
ip device tracking probe auto-source [fallback <i>host-ip-address subnet-mask</i>] [override]	ipv6 neighbor tracking auto-source [fallback <i>host-ip-address subnet-mask</i>] [override]	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking trace-buffer	Not supported	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking maximum n	ipv6 snooping policy IPDT_MAX_n [limit <i>address-count</i>]	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking maximum 0	Not supported	No change, same as Cisco IOS XE Denali 16.3.6
clear ip device tracking all	Not supported	No change, same as Cisco IOS XE Denali 16.3.6

⁹ Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.x, the system incorrectly converts the **ip device tracking probe delay** command to **ipv6 neighbor binding reachable-lifetime**. Starting from Cisco IOS XE Denali 16.3.7 (except Cisco IOS XE Everest 16.5.x), this is corrected to be set to the default value and cannot be changed.

¹⁰ Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.x, the system incorrectly converts the **ip device tracking probe interval** command to **ipv6 snooping tracking retry-interval**. Starting from Cisco IOS XE Denali 16.3.7 (except in Cisco IOS XE Everest 16.5.x), this is correctly converted to **ipv6 neighbor binding reachable-lifetime**.

Table 30: IPDT → SISF Commands

Legacy IPDT	SISF-Based Device-Tracking After SISF Conversion (Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.1a)	SISF-Based Device-Tracking After SISF Conversion (Starting from Cisco IOS XE Denali 16.3.7 and all later releases except Cisco IOS XE Everest 16.5.1a).
ip device tracking probe count	Set to the default value, and cannot be changed.	Set to the default value, and cannot be changed.

Legacy IPDT	SISF-Based Device-Tracking After SISF Conversion (Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.1a)	SISF-Based Device-Tracking After SISF Conversion (Starting from Cisco IOS XE Denali 16.3.7 and all later releases except Cisco IOS XE Everest 16.5.1a).
ip device tracking probe delay	device-tracking binding reachable-lifetime Attention Incorrect system conversion. ¹¹	Set to the default value, and cannot be changed.
ip device tracking probe interval	device-tracking tracking retry-interval Attention Incorrect system conversion. ¹²	device-tracking binding reachable-lifetime
ip device tracking probe use-svi	Set to the default behaviour and cannot be changed.	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking probe auto-source [fallback host-ip-address subnet-mask] [override]	device-tracking tracking auto-source [fallback host-ip-address subnet-mask] [override]	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking trace-buffer	Not supported	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking maximum n	device-tracking snooping policy IPDT_MAX_n [limit address-count]	No change, same as Cisco IOS XE Denali 16.3.6
ip device tracking maximum 0	Not supported	No change, same as Cisco IOS XE Denali 16.3.6
clear ip device tracking all	Not supported	No change, same as Cisco IOS XE Denali 16.3.6

¹¹ Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.x, the system incorrectly converts the **ip device tracking probe delay** command to **device-tracking binding reachable-lifetime**. Starting from Cisco IOS XE Denali 16.3.7 (except in Cisco IOS XE Everest 16.5.x), this is corrected to be set to the default value and cannot be changed.

¹² Until Cisco IOS XE Denali 16.3.6 and in Cisco IOS XE Everest 16.5.x, the system incorrectly converts the **ip device tracking probe interval** command to **device-tracking tracking retry-interval**. Starting from Cisco IOS XE Denali 16.3.7 (except in Cisco IOS XE Everest 16.5.x), this is correctly converted to **device-tracking binding reachable-lifetime**.

How to Configure SISF-Based Device Tracking

Manually Enabling SISF-Based Device Tracking

Applying the Default Device Tracking Policy to a Target

Beginning in privileged EXEC mode, follow these steps to apply the default device tracking policy to an interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	Specify an interface or a VLAN <ul style="list-style-type: none"> • interface <i>interface</i> • vlan configuration <i>vlan_list</i> Example: Device(config)# interface gigabitethernet 1/1/4 OR Device(config)# vlan configuration 333	interface <i>type number</i> —Specifies the interface and enters the interface configuration mode. The device tracking policy will be attached to the specified interface. vlan configuration <i>vlan_list</i> —Specifies the VLANs and enters the VLAN feature configuration mode. The device tracking policy will be attached to the specified VLAN.
Step 3	device-tracking Example: Device(config-if)# device-tracking OR Device(config-vlan-config)# device-tracking	Enables SISF-based device tracking and attaches the default policy it to the interface or VLAN. The default policy is a built-in policy with default settings; none of the attributes of the default policy can be changed.
Step 4	exit Example: Device(config-if)# exit OR Device(config-vlan-config)# exit	Exits configuration mode.
Step 5	show device-tracking policy <i>policy-name</i> Example: Device# show device-tracking policy default	Displays device-tracking policy configuration, and all the targets it is applied to.

Creating a Custom Device Tracking Policy with Custom Settings

Beginning in privileged EXEC mode, follow these steps to create and configure a device tracking policy:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	[no] device-tracking policy <i>policy-name</i> Example: Device (config)# device-tracking policy example_policy	Creates the policy and enters the device-tracking configuration mode.
Step 3	[data-glean default destination-glean device-role distribution-switch exit limit no prefix-glean protocol security-level tracking trusted-port vpc] Example: Device (config-device-tracking)# destination-glean log-only	<p>Enter the question mark (?) at the system prompt to obtain a list of available options in this mode. You can configure the following for both IPv4 and IPv6:</p> <ul style="list-style-type: none"> • (Optional) data-glean—Enables learning of addresses from a data packet snooped from a source inside the network and populates the binding table with the data traffic source address. Enter one of these options: <ul style="list-style-type: none"> • log-only—Generates a syslog message upon data packet notification • recovery—Uses a protocol to enable binding table recovery. Enter NDP or DHCP. • (Optional) default—Sets the policy attribute to its default value. You can set these policy attributes to their default values: data-glean, destination-glean, device-role, limit, prefix-glean, protocol, security-level, tracking, trusted-port. • (Optional) destination-glean—Populates the binding table by gleaning data traffic destination address. Enter one of these options: <ul style="list-style-type: none"> • log-only—Generates a syslog message upon data packet notification • recovery—Uses a protocol to enable binding table recovery. Enter DHCP.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) device-role—Sets the role of the device attached to the port. It can be a node or a switch. Enter one of these options: <ul style="list-style-type: none"> • node—Configures the attached device as a node. This is the default option. • switch—Configures the attached device as a switch. • (Optional) distribution-switch—Although visible on the CLI, this option is not supported. Any configuration settings you make will not take effect. • exit—Exits the device-tracking policy configuration mode. • limit address-count—Specifies an address count limit per port. The range is 1 to 32000. • no—Negates the command or sets it to defaults. • (Optional) prefix-glean—Enables learning of prefixes from either IPv6 Router Advertisements or from DHCP-PD. You have the following option: <ul style="list-style-type: none"> • (Optional) only—Gleans only prefixes and not host addresses. • (Optional) protocol—Sets the protocol to glean; by default, all are gleaned. Enter one of these options: <ul style="list-style-type: none"> • arp [prefix-list name]—Gleans addresses in ARP packets. Optionally, enter the name of prefix-list that is to be matched. • dhcp4 [prefix-list name]—Glean addresses in DHCPv4 packets. Optionally, enter the name of prefix-list that is to be matched. • dhcp6 [prefix-list name]—Glean addresses in DHCPv6 packets. Optionally, enter the name of prefix-list that is to be matched.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • ndp [prefix-list <i>name</i>]—Glean addresses in NDP packets. Optionally, enter the name of prefix-list that is to be matched. • udp [prefix-list <i>name</i>]—Although visible on the CLI, this option is not supported. Any configuration settings you make will not take effect. • (Optional) security-level—Specifies the level of security enforced by the feature. Enter one of these options: <ul style="list-style-type: none"> • glean—Gleans addresses passively. • guard—Inspects and drops un-authorized messages. This is the default. • inspect—Gleans and validates messages. • (Optional) tracking—Specifies a tracking option. Enter one of these options: <ul style="list-style-type: none"> • disable [stale-lifetime [<i>1-86400-seconds</i> infinite]] —Turns off device-tracking. Optionally, you can enter the duration for which the entry is kept inactive before deletion, or keep it permanently inactive. • enable [reachable-lifetime [<i>1-86400-seconds</i> infinite]] —Turns on device-tracking. Optionally, you can enter the duration for which the entry is kept reachable, or keep it permanently reachable. • (Optional) trusted-port—Sets up a trusted port. Disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table. • (Optional) vpc—Although visible on the CLI, this option is not supported. Any

	Command or Action	Purpose
		configuration settings you make will not take effect.
Step 4	end Example: Device(config-device-tracking)# exit	Exits configuration mode.
Step 5	show device-tracking policy <i>policy-name</i> Example: Device# show device-tracking policy example_policy	Displays the device-tracking policy configuration.

What to do next

Attach the policy to an interface or VLAN.

Attaching a Device Tracking Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach a device tracking policy to an interface:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface <i>interface</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface and enters the interface configuration mode.
Step 3	[no] device-tracking attach-policy <i>policy name</i> Example: Device(config-if)# device-tracking attach-policy example_policy	Attaches the device tracking policy to the interface. Note SISF based device-tracking policies can be disabled only if they are custom policies. Programmatically created policies can be removed only if the corresponding device-tracking client feature configuration is removed.
Step 4	end Example: Device# end	Returns to the privileged EXEC mode.

	Command or Action	Purpose
Step 5	show device-tracking policies [interface interface] Example: Device# show device-tracking policies interface gigabitethernet 1/1/4	Displays policies that match the specified interface type and number.

Attaching a Device Tracking Policy to a VLAN

Beginning in privileged EXEC mode, follow these steps to attach a device-tracking policy to VLANs across multiple interfaces:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	vlan configuration vlan_list Example: Device(config)# vlan configuration 333	Specifies the VLANs to which the device tracking policy will be attached; enters the VLAN interface configuration mode.
Step 3	[no] device-tracking attach-policy policy_name Example: Device(config-vlan-config)# device-tracking attach-policy example_policy	Attaches the device tracking policy to the specified VLANs across all switch interfaces. Note SISF based device-tracking policies can be disabled only if they are custom policies. Programmatically created policies can be removed only if the corresponding device-tracking client feature configuration is removed.
Step 4	do show device-tracking policies vlan vlan-ID Example: Device(config-vlan-config)# do show device-tracking policies vlan 333	Verifies that the policy is attached to the specified VLAN, without exiting the VLAN interface configuration mode.

Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE Everest 16.5.x

Table 31: Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE Everest 16.5.x

Device tracking client features that can enable SISF-based device tracking	<p>In this release, you can programmatically enable SISF-based device tracking for these features:</p> <ul style="list-style-type: none"> • IEEE 802.1X, web authentication, Cisco TrustSec, and IPSG features: enter the ip dhcp snooping vlan <i>vlan</i> command. • Cisco Locator/ID Separation Protocol. <p>Note LISP settings are effective, if there is more than one programmatically created policy. This does not adversely affect the way, any of the other device tracking client features work. For example, if you have configured the ip dhcp snooping vlan <i>vlan</i> command for IEEE 802.1X, and also enabled SISF-based device tracking by configuring LISP, the IEEE 802.1X feature continues to work as expected.</p>
Policy Name	<p>DT-PROGRAMMATIC</p> <p>Although there is more than one device tracking client feature, the system-generated policy is one and the same; the list of settings may differ with each programmatically created policy. See the examples for more information.</p>
User Options	<ul style="list-style-type: none"> • Only one policy can be attached to the same interface or VLAN. • The policy cannot be replaced by another policy. • The policy cannot be removed unless the device tracking client feature configuration is removed. • The policy attributes cannot be changed. • The address count limit per MAC setting cannot be changed (This refers to the limit address-count for IPv4 per mac and limit address-count for IPv6 per mac commands), but the address count limit per port or interface can be changed. • When a device-tracking policy is attached to an interface under a VLAN, the policy settings on the interface take precedence over those on its VLAN; exceptions here are the values for limit address-count for IPv4 per mac and limit address-count for IPv6 per mac, which are aggregated from the policy on both the interface and VLAN.

Configuring a Multi-Switch Network to Stop Creating Binding Entries from a Trunk Port

In a multi-switch network, SISF-based device tracking provides the capability to distribute binding table entries between switches running the feature. Binding entries are only created on the switches where the host

appears on an access port. No entry is created for a host that appears over a trunk port. This is achieved by configuring a policy with the **trusted-port** and **device-role switch** options, and attaching it to the trunk port.



Important

Both, the **trusted-port**, and **device-role switch** options, must be configured in the policy.

Further, we recommended that you apply such a policy on a port facing a device, which also has SISF-based device tracking enabled.

Complete the following steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	device-tracking policy <i>policy-name</i> Example: Device(config)# device-tracking policy example_trusted_policy	Enters the device-tracking policy configuration mode, for the specified policy.
Step 3	device-role switch Example: Device(config-device-tracking)# device-role switch	Specifies the role of the device attached to the port. Default is node. Enter the device-role switch option to stop the creation of binding entries for the port.
Step 4	trusted-port Example: Device(config-device-tracking)# trusted-port	Sets up a trusted port. Disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 5	end Example: Device(config-device-tracking)# end	Exits the device-tracking policy configuration mode and enters the global configuration mode
Step 6	interface <i>interface</i> Example: Device(config)# interface gigabitethernet 1/0/25	Specifies an interface and enters the interface configuration mode.
Step 7	device-tracking attach-policy <i>policy-name</i> Example: Device(config-if)# device-tracking attach-policy example_trusted_policy	Attaches a device tracking policy to the interface or the specified VLANs on the interface.

Configuration Examples for SISF-Based Device Tracking

These examples show sample device-tracking configuration and other recommended or related configuration for certain situations.

Example: Programmatically Enabling SISF-Based Device Tracking in Cisco IOS XE Everest 16.5.x

In this release, SISF-Based device-tracking can be enabled by entering the **ip dhcp snooping vlan** *vlan* command OR by configuring LISP and the examples here show both ways.

Device tracking client: LISP

After you configure LISP, enter the **show device-tracking policy** command in privileged EXEC mode, to display the `LISP-DT-GUARD-VLAN` policy that is created and the corresponding settings.

```
Device(config)# router lisp
<output truncated>
Device(config-router-lisp)# instance-id 3
Device(config-router-lisp-instance)# service ethernet
Device(config-router-lisp-instance-service)# eid-table vlan 10
Device(config-router-lisp-instance-dynamic-eid)# database-mapping 10.1.1.0/24 locator-set
set1
Device(config-router-lisp-instance-service)# exit-service-ethernet
Device(config-router-lisp-instance)# exit-instance-id
Device(config-router-lisp)# exit-router-lisp

Device# show device-tracking policy DT-PROGRAMMATIC
Policy DT-PROGRAMMATIC configuration:
  security-level guard (*)
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  tracking enable
Policy DT-PROGRAMMATIC is applied on the following targets:
Target      Type  Policy          Feature          Target range
vlan 10     VLAN  DT-PROGRAMMATIC  Device-tracking  vlan all
note:
Binding entry Down timer: 10 minutes (*)
Binding entry Stale timer: 60 minutes (*)
```

Device tracking clients: IEEE 802.1X, Web Authentication, Cisco TrustSec, IPSG

Configure the **ip dhcp snooping vlan** *vlan* command in global configuration mode to enable device-tracking for the IEEE 802.1X, web authentication, Cisco TrustSec, and IPSP features. Enter the **show device-tracking policy** command in privileged EXEC mode, to display the `DT-PROGRAMMATIC` policy that is created and the corresponding settings that are made:

```
Device# configure terminal
Device(config)# ip dhcp snooping vlan 10
```

Example: Disabling IPv6 Device Tracking on a Target

```

Device(config)# end

Device# show device-tracking policy DT-PROGRAMMATIC
Policy DT-PROGRAMMATIC configuration:
  security-level glean (*)
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  tracking enable
Policy DT-PROGRAMMATIC is applied on the following targets:
Target  Type  Policy  Feature  Target range
vlan 10  VLAN  DT-PROGRAMMATIC  Device-tracking  vlan all
note:
  Binding entry Down timer: 24 hours (*)
  Binding entry Stale timer: 24 hours (*)

```

Example: Disabling IPv6 Device Tracking on a Target

By default, SISF-based device tracking supports both IPv4 and IPv6. The following configuration examples show how you can disable IPv6 device tracking if you have to:

Disabling IPv6 device tracking when the target is attached to a custom policy:

```

Device(config)# device-tracking policy example-policy
Device(config-device-tracking)# no protocol ndp
Device(config-device-tracking)# no protocol dhcp6
Device(config-device-tracking)# end

```



Note In the Cisco IOS XE Denali 16.3.x and Cisco IOS XE Everest 16.5.x releases, you cannot disable IPv6 device tracking for a programmatically created policy.

Example: Enabling IPv6 for SVI on VLAN (To Mitigate the Duplicate Address Problem)

When IPv6 is enabled in the network and a switched virtual interface (SVI) is configured on a VLAN, we recommend that you add the following to the SVI configuration. This enables the SVI to acquire a link-local address automatically; this address is used as the source IP address of the SISF probe, thus preventing the duplicate IP address issue.

```

Device(config)# interface vlan 10
Device(config-if)# ipv6 enable
Device(config-if)# end

```

Example: Mitigating the IPv4 Duplicate Address Problem

This example show how you can tackle the Duplicate IP Address 0.0.0.0 error message problem encountered by clients that run Microsoft Windows:

Configure the **device-tracking tracking auto-source** command. This command determines the source IP and MAC address used in the Address Resolution Packet (ARP) request sent by the switch to probe a client, in order to maintain its entry in the device-tracking table. The purpose, is to avoid using 0.0.0.0 as source IP address.



Note Configure the **device-tracking tracking auto-source** command only when a switch virtual interface (SVI) is not configured. You do not have to configure it when a SVI is configured with an IPv4 address on the VLAN.

Command	Action (In order to select source IP and MAC address for device tracking ARP probe)	Notes
device-tracking tracking auto-source	<ul style="list-style-type: none"> • Set source to VLAN SVI if present. • Look for IP and MAC binding in device-tracking table from same subnet. • Use 0.0.0.0 	We recommend that you disable device-tracking on all trunk ports to avoid MAC flapping.
device-tracking tracking auto-source override	<ul style="list-style-type: none"> • Set source to VLAN SVI if present • Use 0.0.0.0 	Not recommended when there is no SVI.
device-tracking tracking auto-source fallback 0.0.0.X 255.255.255.0	<ul style="list-style-type: none"> • Set source to VLAN SVI if present. • Look for IP and MAC binding in device-tracking table from same subnet. • Compute source IP from client IP using host bit and mask provided. Source MAC is taken from the MAC address of the switchport facing the client*. 	<p>We recommend that you disable device-tracking on all trunk ports to avoid MAC flapping.</p> <p>The computed IPv4 address must not be assigned to any client or network device.</p>

Command	Action (In order to select source IP and MAC address for device tracking ARP probe)	Notes
device-tracking tracking auto-source fallback 0.0.0.X 255.255.255.0 override	<ul style="list-style-type: none"> Set source to VLAN SVI if present. Compute source IP from client IP using host bit and mask provided*. Source MAC is taken from the MAC address of the switchport facing the client*.	

* Depending on the client IP address, an IPv4 address has to be reserved for the source IP.

A reserved source IPv4 address = (client-ip and mask) | host-ip

- Client IP = 192.0.2.25
- Source IP = (192.0.2.25 and 255.255.255.0) | (0.0.0.1) = 192.0.2.1

IP address 192.0.2.1 should not be assigned to any client or network device.

Example: Avoiding a Short Device-Tracking Binding Reachable Time

When migrating from an older release, the following configuration may be present:

```
device-tracking binding reachable-time 10
```

Remove this by entering the **no** version of the command.

Feature History and Information for SISF-Based Device Tracking

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Release	Modification
Cisco IOS XE Denali 16.1.1	This feature was introduced.

Release	Modification
Cisco IOS XE Denali 16.3.7	<p>Correction in the system conversion of IPv6 snooping commands and SISF-based device-tracking commands.</p> <p>IPDT → IPv6 Snooping conversion corrections:</p> <ul style="list-style-type: none"> • Until Cisco IOS XE Denali 16.3.6, the system incorrectly converts the ip device tracking probe delay command to ipv6 neighbor binding reachable-lifetime. Starting from Cisco IOS XE Denali 16.3.7, this is corrected to be set to the default value and cannot be changed. • Until Cisco IOS XE Denali 16.3.6, the system incorrectly converts the ip device tracking probe interval command to ipv6 snooping tracking retry-interval. Starting from Cisco IOS XE Denali 16.3.7, this is correctly converted to ipv6 neighbor binding reachable-lifetime. <p>IPDT → SISF conversion corrections:</p> <ul style="list-style-type: none"> • Until Cisco IOS XE Denali 16.3.6, the system incorrectly converts the ip device tracking probe delay command to device-tracking binding reachable-lifetime. Starting from Cisco IOS XE Denali 16.3.7, this is corrected to be set to the default value and cannot be changed. • Until Cisco IOS XE Denali 16.3.6, the system incorrectly converts the ip device tracking probe interval command to device-tracking tracking retry-interval. Starting from Cisco IOS XE Denali 16.3.7, this is correctly converted to device-tracking binding reachable-lifetime.
Cisco IOS XE Everest 16.5.1a	<p>The feature can now be programmatically enabled, by configuring LISP.</p> <p>The system-generated policy name is changed to DT-PROGRAMMATIC. Although there is more than device tracking client, the system-generated policy is one and the same; the list of settings differ depending on the device tracking client that enables SISF.</p> <p>You can also create a custom policy and attach it to an interface or VLAN.</p> <p>Although the corrections in the system conversion of IPv6 snooping commands and SISF-based device-tracking commands are available starting from Cisco IOS XE Denali 16.3.7, these corrections are NOT available in Cisco IOS XE Everest 16.5.x.</p>



CHAPTER 18

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

- [Finding Feature Information, on page 325](#)
- [Information About 802.1x Port-Based Authentication, on page 325](#)
- [How to Configure 802.1x Port-Based Authentication, on page 356](#)
- [Monitoring 802.1x Statistics and Status, on page 404](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.



Note TACACS is not supported with 802.1x authentication.

Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

The table shown below lists the maximum number of each client session supported on Catalyst 3850 and Catalyst 3650 switches:

Client session	Maximum sessions supported
Maximum dot1x or MAB client sessions	2000
Maximum web-based authentication sessions	2000
Maximum dot1x sessions with critical-auth VLAN enabled and server re-initialized	2000
Maximum MAB sessions with various session features applied	2000
Maximum dot1x sessions with service templates or session features applied	2000



Note To view debug logs for RADIUS and AAA, use the **show platform software trace message smd** command. For more information, see the Trace Commands section in *Command Reference Guide, Cisco IOS XE Denali 16.1.1*

Port-Based Authentication Process

To configure IEEE 802.1X port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

The AAA process begins with authentication. When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

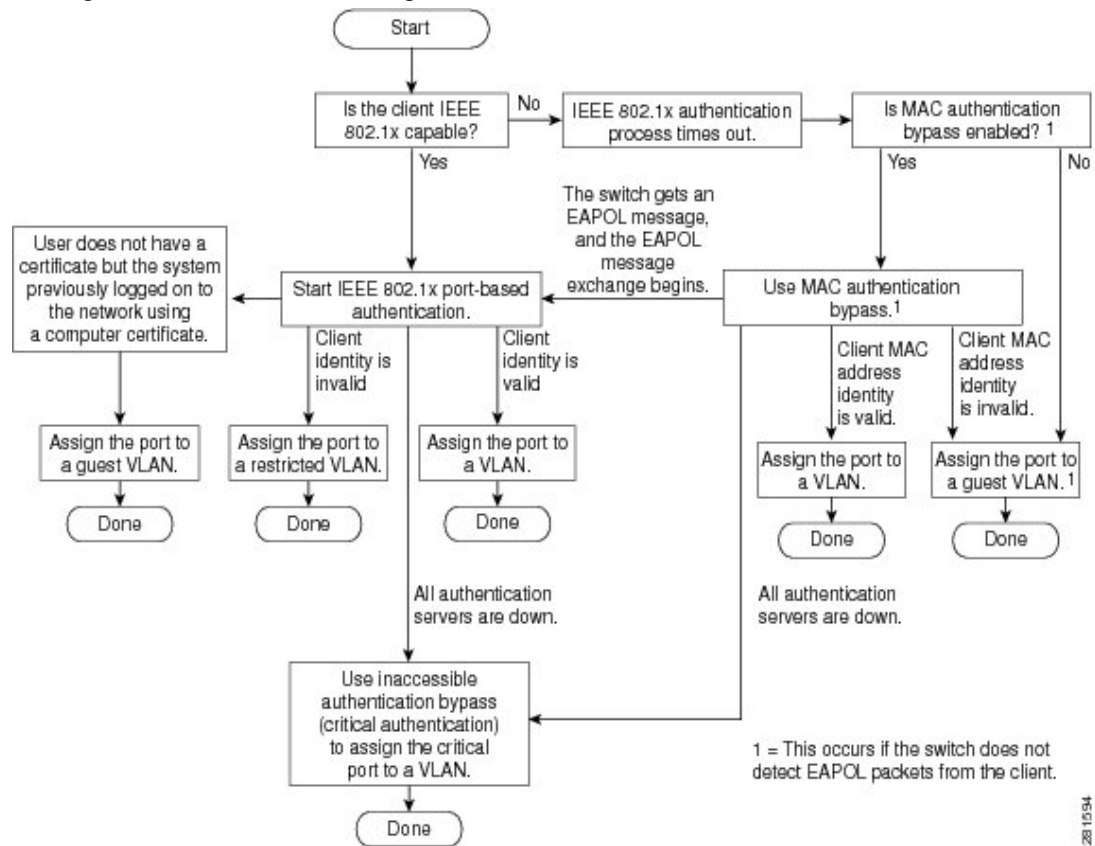


Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

Figure 21: Authentication Flowchart

This figure shows the authentication process.



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs. The range is 1 to 65535 seconds.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the

attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is *RADIUS-Request*), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **dot1x re-authenticate interface *interface-id*** privileged EXEC command.

Port-Based Authentication Initiation and Message Exchange

During 802.1x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.



Note

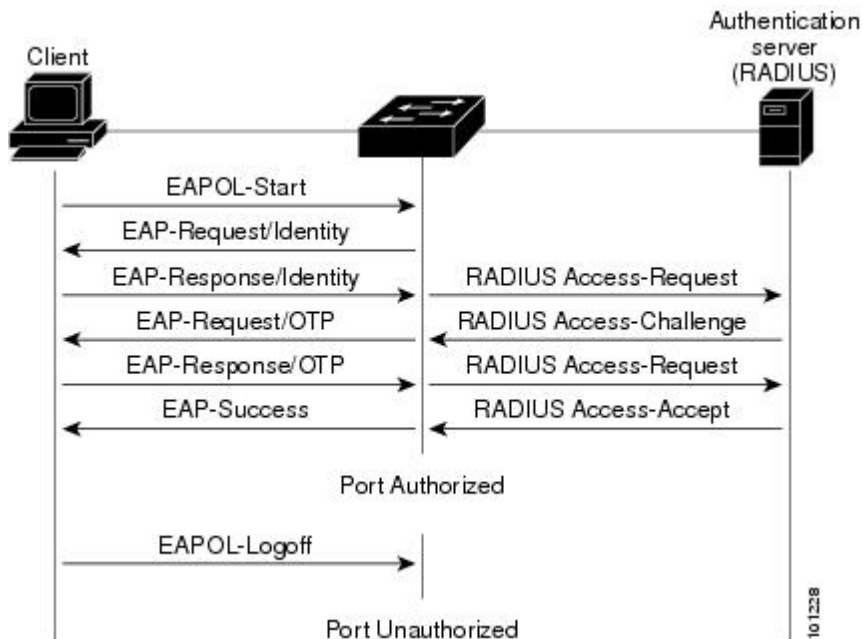
If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted.

The specific exchange of EAP frames depends on the authentication method being used.

Figure 22: Message Exchange

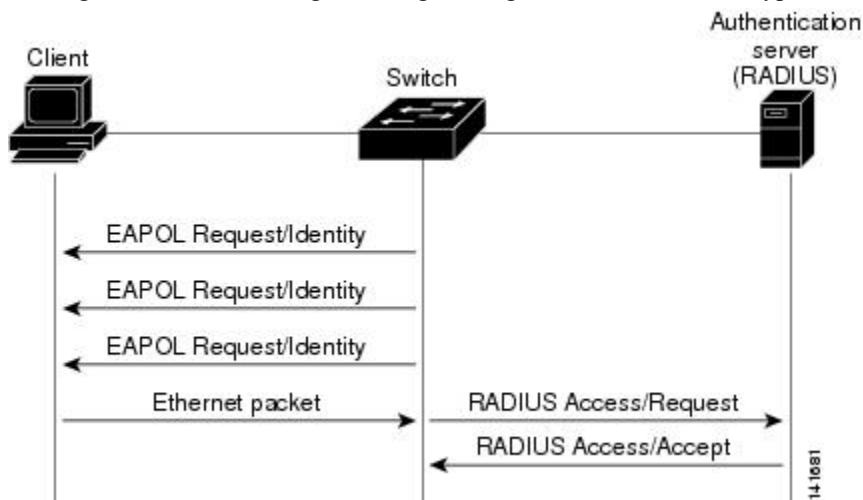
This figure shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.



If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and starts 802.1x authentication.

Figure 23: Message Exchange During MAC Authentication Bypass

This figure shows the message exchange during MAC authentication bypass.



Authentication Manager for Port-Based Authentication

Port-Based Authentication Methods

Table 32: 802.1x Features

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL			
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL
Web authentication as fallback method	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

¹³ Supported in Cisco IOS Release 12.2(50)SE and later.

¹⁴ For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids



Note Using role-based ACLs as Filter-Id is not recommended.

More than one host can be authenticated on MDA-enabled and multiauth ports. The ACL policy applied for one host does not effect the traffic of another host. If only one host is authenticated on a multi-host port, and

the other hosts gain network access without authentication, the ACL policy for the first host can be applied to the other connected hosts by specifying any in the source address.

Port-Based Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands begin with the **dot1x** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface.

To disable dot1x on a switch, remove the configuration globally by using the **no dot1x system-auth-control** command, and also remove it from all configured interfaces.



Note If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

To re-enable dot1x on the switch, you must configure both the dot1x global and interface configurations. Incomplete configurations can cause high CPU utilization.

The **authentication manager** commands provide the same functionality as earlier 802.1x commands.

When filtering out verbose system messages generated by the authentication manager, the filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Table 33: Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction {both in}	dot1x control-direction {both in}	Enable 802.1x authentication with the wake-on-LAN (WoL) feature, and configure the port control as unidirectional or bidirectional.

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication event	dot1x auth-fail vlan dot1x critical (interface configuration) dot1x guest-vlan6	Enable the restricted VLAN on a port. Enable the inaccessible-authentication-bypass feature. Specify an active VLAN as an 802.1x guest VLAN.
authentication fallback <i>fallback-profile</i>	dot1x fallback <i>fallback-profile</i>	Configure a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
authentication host-mode [multi-auth multi-domain multi-host single-host]	dot1x host-mode { single-host multi-host multi-domain }	Allow a single host (client) or multiple hosts on an 802.1x-authorized port.
authentication order	mab	Provides the flexibility to define the order of authentication methods to be used.
authentication periodic	dot1x reauthentication	Enable periodic re-authentication of the client.
authentication port-control { auto force-authorized force-unauthorized }	dot1x port-control { auto force-authorized force-unauthorized }	Enable manual control of the authorization state of the port.
authentication timer	dot1x timeout	Set the 802.1x timers.
authentication violation { protect restrict shutdown }	dot1x violation-mode { shutdown restrict protect }	Configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.



Note CDP bypass is not supported and may cause a port to go into err-disabled state.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized**—disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto**—enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

Port-Based Authentication and Switch Stacks

If a switch is added to or removed from a switch stack, 802.1x authentication is not affected as long as the IP connectivity between the RADIUS server and the stack remains intact. This statement also applies if the stack's active switch is removed from the switch stack. Note that if the active switch fails, a stack member becomes the new active switch of the stack by using the election process, and the 802.1x authentication process continues as usual.

If IP connectivity to the RADIUS server is interrupted because the switch that was connected to the server is removed or fails, these events occur:

- Ports that are already authenticated and that do not have periodic re-authentication enabled remain in the authenticated state. Communication with the RADIUS server is not required.

- Ports that are already authenticated and that have periodic re-authentication enabled (with the **dot1x re-authentication** global configuration command) fail the authentication process when the re-authentication occurs. Ports return to the unauthenticated state during the re-authentication process. Communication with the RADIUS server is required.

For an ongoing authentication, the authentication fails immediately because there is no server connectivity.

If the switch that failed comes up and rejoins the switch stack, the authentications might or might not fail depending on the boot-up time and whether the connectivity to the RADIUS server is re-established by the time the authentication is attempted.

To avoid loss of connectivity to the RADIUS server, you should ensure that there is a redundant connection to it. For example, you can have a redundant connection to the stack's active switch and another to a stack member, and if the active switch fails, the switch stack still has connectivity to the RADIUS server.

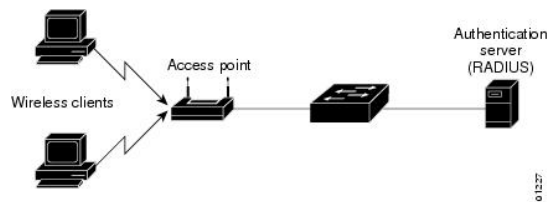
802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode, only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients.

In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

Figure 24: Multiple Host Mode Example



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

The switch supports multidomain authentication (MDA), which allows both a data device and a voice device, such as an IP Phone (Cisco or non-Cisco), to connect to the same switch port.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN and voice VLAN. Each host is individually authenticated. There is no limit to the number of data or voice device that can be authenticated on a multiauthport.

If a hub or access point is connected to an 802.1x-enabled port, each connected client must be authenticated. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the per-host authentication fallback method to authenticate different hosts with different methods on a single port.



Note When a port is in multiple-authentication mode, the authentication-failed VLAN features do not activate.

You can assign a RADIUS-server-supplied VLAN in multi-auth mode, under the following conditions:

- The host is the first host authorized on the port, and the RADIUS server supplies VLAN information
- Subsequent hosts are authorized with a VLAN that matches the operational VLAN.
- A host is authorized on the port with no VLAN assignment, and subsequent hosts either have no VLAN assignment, or their VLAN information matches the operational VLAN.
- The first host authorized on the port has a group VLAN assignment, and subsequent hosts either have no VLAN assignment, or their group VLAN matches the group VLAN on the port. Subsequent hosts must use the same VLAN from the VLAN group as the first host. If a VLAN list is used, all hosts are subject to the conditions specified in the VLAN list.
- After a VLAN is assigned to a host on the port, subsequent hosts must have matching VLAN information or be denied access to the port.
- You cannot configure a guest VLAN or an auth-fail VLAN in multi-auth mode.
- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

Multi-auth Per User VLAN assignment

The Multi-auth Per User VLAN assignment feature allows you to create multiple operational access VLANs based on VLANs assigned to the clients on the port that has a single configured access VLAN. The port configured as an access port where the traffic for all the VLANs associated with data domain is not dot1q tagged, and these VLANs are treated as native VLANs.

The number of hosts per multi-auth port is 8, however there can be more hosts.

The following scenarios are associated with the multi-auth Per User VLAN assignments:

Scenario one

When a hub is connected to an access port, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. This behaviour is similar on a single-host or multi-domain-auth port.

When a second host (H2) is connected and gets assigned to VLAN (V2), the port will have two operational VLANs (V1 and V2). If H1 and H2 sends untagged ingress traffic, H1 traffic is mapped to VLAN (V1) and H2 traffic to VLAN (V2), all egress traffic going out of the port on VLAN (V1) and VLAN (V2) are untagged.

If both the hosts, H1 and H2 are logged out or the sessions are removed due to some reason then VLAN (V1) and VLAN (V2) are removed from the port, and the configured VLAN (V0) is restored on the port.

Scenario two

When a hub is connected to an access port, and the port is configured with an access VLAN (V0). The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1.

When a second host (H2) is connected and gets authorized without explicit vlan policy, H2 is expected to use the configured VLAN (V0) that is restored on the port. All egress traffic going out of two operational VLANs, VLAN (V0) and VLAN (V1) are untagged.

If host (H2) is logged out or the session is removed due to some reason then the configured VLAN (V0) is removed from the port, and VLAN (V1) becomes the only operational VLAN on the port.

Scenario three

When a hub is connected to an access port in open mode, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. When a second host (H2) is connected and remains unauthorized, it still has access to operational VLAN (V1) due to open mode.

If host H1 is logged out or the session is removed due to some reason, VLAN (V1) is removed from the port and host (H2) gets assigned to VLAN (V0).



Note The combination of Open mode and VLAN assignment has an adverse affect on host (H2) because it has an IP address in the subnet that corresponds to VLAN (V1).

Limitation in Multi-auth Per User VLAN assignment

In the Multi-auth Per User VLAN assignment feature, egress traffic from multiple vlans are untagged on a port where the hosts receive traffic that is not meant for them. This can be a problem with broadcast and multicast traffic.

- **IPv4 ARPs:** Hosts receive ARP packets from other subnets. This is a problem if two subnets in different Virtual Routing and Forwarding (VRF) tables with overlapping IP address range are active on the port. The host ARP cache may get invalid entries.
- **IPv6 control packets:** In IPv6 deployments, Router Advertisements (RA) are processed by hosts that are not supposed to receive them. When a host from one VLAN receives RA from a different VLAN, the host assign incorrect IPv6 address to itself. Such a host is unable to get access to the network.

The workaround is to enable the IPv6 first hop security so that the broadcast ICMPv6 packets are converted to unicast and sent out from multi-auth enabled ports.. The packet is replicated for each client in multi-auth port belonging to the VLAN and the destination MAC is set to an individual client. Ports having one VLAN, ICMPv6 packets broadcast normally.

- **IP multicast:** Multicast traffic destined to a multicast group gets replicated for different VLANs if the hosts on those VLANs join the multicast group. When two hosts in different VLANs join a multicast group (on the same mutli-auth port), two copies of each multicast packet are sent out from that port.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port. MAC move is supported on all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the that port.) When a MAC address moves from one port to another, the switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port. The MAC move feature applies to both voice and data hosts.



Note In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

MAC Replace

The MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.



Note This feature does not apply to ports in multi-auth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multi-domain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.
- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.

- Link-down occurs.
- Re-authentication successfully occurs.
- Re-authentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START—sent when a new user session starts
- INTERIM—sent during an existing session for updates
- STOP—sent when a session terminates



Note

To view debug logs for RADIUS and AAA, use the **show platform software trace message smd** command. For more information, see the Tracing Commands section in *Command Reference Guide, Cisco IOS XE Denali 16.1.1*.

This table lists the AV pairs and when they are sent are sent by the switch.

Table 34: Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes ¹⁵	Sometimes
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[47]	Acct-Input-Packets	Never	Always	Always
Attribute[48]	Acct-Output-Packets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

¹⁵ The Framed-IP-Address AV pair is sent when a valid static IP address is configured or when a Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for the devices that do not support 802.1x functionality.

This feature only works if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After successful 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

Voice device authentication is supported with multidomain host mode in Cisco IOS Release 12.2(37)SE. In Cisco IOS Release 12.2(40)SE and later, when a voice device is authorized and the RADIUS server returned an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports.

When configured on the switch and the RADIUS server, 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, a shut down or suspended VLAN. In the case of a multidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse).

- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN and configured voice VLAN.
- If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:
 - If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, then authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
 - If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access VLAN.

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
 - [64] Tunnel-Type = VLAN
 - [65] Tunnel-Medium-Type = 802

- [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID
- [83] Tunnel-Preference

Attribute [64] must contain the value *VLAN* (type 13). Attribute [65] must contain the value *802* (type 6). Attribute [81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. Incoming packets received on the port to which a port ACL is applied are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inac1#<n>` for the ingress direction and `outac1#<n>` for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports.

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the `Filter-Id` attribute, it can point to a standard ACL.

You can use the `Filter-Id` attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by `.in` for ingress filtering or `.out` for egress filtering. If the RADIUS server does not allow the `.in` or `.out` syntax, the access list is applied to the outbound ACL by default. Because of limited support of Cisco IOS access lists on the switch, the `Filter-Id` attribute is supported only for IP ACLs numbered 1 to 199 and 1300 to 2699 (IP standard and IP extended ACLs).

The maximum size of the per-user ACL is 4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

To configure per-user ACLs:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.



Note Per-user ACLs are supported only in single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.



Note A downloadable ACL is also referred to as a *dACL*.

If more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.



Note The limit for dACL with stacking is 64 ACEs per dACL per port. The limit without stacking is the number of available TCAM entries which varies based on the other ACL features that are active.

For a URL redirect ACL:

- Packets that match a permit access control entry (ACE) rule are sent to the CPU for forwarding to the AAA server.
- Packets that match a deny ACE rule are forwarded through the switch.
- Packets that match neither the permit ACE rule or deny ACE rule are processed by the next dACL, and if there is no dACL, the packets hit the implicit-deny ACL and are dropped.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP or HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-defined-ACL attribute value pair to intercept an HTTP or HTTPS request from the end point. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect.

**Note**

- Traffic that matches a permit ACE in the ACL is redirected.
- An ACE that matches permit rule of the url-redirect-acl gets the client redirected to url-redirect page. The client traffic is allowed when a deny rule is matched.
- Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

When security ACL/dACL and punt/redirect ACLs are applied together to the session, the url-redirect-acl has the higher priority.

For more information about using redirect ACLs, refer the document [here](#).

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value (AV) pair on the Cisco Secure ACS with the RADIUS cisco-av-pair vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the #ACL#-IP-name-number attribute.

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a host-access-policy to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives an host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

VLAN ID-based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

**Note**

This feature is not supported on Cisco ACS Server. (The ACS server ignores the sent VLAN-IDs for new hosts and only authenticates based on the MAC address.)

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.



Note

If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

Any number of 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single host, multiple host, multi-auth and multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when IEEE 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each IEEE 802.1x port on a switch stack or a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.



Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link down* or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported on 802.1x ports in all host modes and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other security port features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy*, when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to *critical ports*.

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the *critical VLAN*. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the *critical-authentication* state, which is a special case of the authentication state.



Note If *critical authentication* is configured on interface, then vlan used for critical authorization (*critical vlan*) should be active on the switch. If the *critical vlan* is inactive (or) down, *critical authentication* session will keep trying to enable inactive vlan and fail repeatedly. This can lead to large amount of memory holding.

Inaccessible Authentication Bypass Support on Multiple-Authentication Ports

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan *vlan-id*** command. When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

This command is supported on all host modes.

Inaccessible Authentication Bypass Authentication Results

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and reauthentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically re-authenticated.

Inaccessible Authentication Bypass Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on 802.1x port, the features interact as follows:
 - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
 - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.

- If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.
- If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on a private VLAN host port. The access VLAN must be a secondary private VLAN.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

In a switch stack:

- The stack's active switch checks the status of the RADIUS servers by sending keepalive packets. When the status of a RADIUS server changes, the stack's active switch sends the information to the stack members. The stack members can then check the status of RADIUS servers when re-authenticating critical ports.
- If the new active switch is elected, the link between the switch stack and RADIUS server might change, and the new stack immediately sends keepalive packets to update the status of the RADIUS servers. If the server status changes from *dead* to *alive*, the switch re-authenticates all switch ports in the critical-authentication state.

When a member is added to the stack, the stack's active switch sends the member the server status.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the access control server (ACS), the phone is put into the voice domain. If the ACS is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.

You can enter the **authentication event server dead action authorize voice** interface configuration command to configure the critical voice VLAN feature. When the ACS does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through CDP (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan *vlan-id*** interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch is in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1x user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found, the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.



Note The RADIUS server can send the VLAN information in any combination of VLAN-IDs, VLAN names, or VLAN groups.

802.1x User Distribution Configuration Guidelines

- Confirm that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the authenticated ports in the VLAN are cleared, but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is cleared.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.

IEEE 802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.
- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of IEEE 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a switch port, you can configure an access port VLAN that is also a voice VLAN.

When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone



Note If you enable IEEE 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

IEEE 802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x authentication with wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an IEEE 802.1x port and the host powers off, the IEEE 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.



Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an IEEE 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured. This process works for most client devices; however, it does not work for clients that use an alternate MAC address format. You can configure how MAB authentication is performed for clients with MAC addresses that deviate from the standard format or where the RADIUS configuration requires the user name and password to differ.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant and uses 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses the authentication or re-authentication methods configured on the port, if the previous session ended because the Termination-Action RADIUS attribute value is *DEFAULT*.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize* (the attribute value is *DEFAULT*), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

MAC authentication bypass interacts with the features:

- IEEE 802.1x authentication—You can enable MAC authentication bypass only if 802.1x authentication is enabled on the port .
- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.
- Restricted VLAN—This feature is not supported when the client connected to an IEEE 802.1x port is authenticated with MAC authentication bypass.
- Port security
- Voice VLAN
- Private VLAN—You can assign a client to a private VLAN.
- Network Edge Access Topology (NEAT)—MAB and NEAT are mutually exclusive. You cannot enable MAB when NEAT is enabled on an interface, and you should not enable NEAT when MAB is enabled on an interface.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 IEEE 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 IEEE 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- Set the list of VLAN number or name or VLAN group name as the value of the Tunnel Group Private ID (Attribute[81]) and the preference for the VLAN number or name or VLAN group name as the value of the Tunnel Preference (Attribute[83]). If you do not configure the Tunnel Preference, the first Tunnel Group Private ID (Attribute[81]) attribute is picked up from the list.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** privileged EXEC command.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. The IEEE 802.1X Flexible Authentication feature supports three authentication methods:

- dot1X—IEEE 802.1X authentication is a Layer 2 authentication method.
- mab—MAC-Authentication Bypass is a Layer 2 authentication method.
- webauth—Web authentication is a Layer 3 authentication method.

Using this feature, you can control which ports use which authentication methods, and you can control the failover sequencing of methods on those ports. For example, MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail.

The IEEE 802.1X Flexible Authentication feature supports the following host modes:

- multi-auth—Multiauthentication allows one authentication on a voice VLAN and multiple authentications on the data VLAN.
- multi-domain—Multidomain authentication allows two authentications: one on the voice VLAN and one on the data VLAN.

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication—Only one user is allowed network access before and after authentication.
- MDA mode with open authentication—Only one user in the voice domain and one user in the data domain are allowed.
- Multiple-hosts mode with open authentication—Any host can access the network.
- Multiple-authentication mode with open authentication—Similar to MDA, except multiple hosts can be authenticated.



Note If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.

Follow these guidelines for configuring MDA:

- You must configure a switch port for MDA.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain.
- Voice VLAN assignment on an MDA-enabled port is supported Cisco IOS Release 12.2(40)SE and later.
- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of *device-traffic-class=voice*. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support IEEE 802.1x authentication.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.
- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.
- When a port host mode is changed from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone that has been allowed on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single- or multihost mode to multidomain mode.
- Switching a port host mode from multidomain to single- or multihost mode removes all authorized devices from the port.

- If a data domain is authorized first and placed in the guest VLAN, non-IEEE 802.1x-capable voice devices need to tag their packets on the voice VLAN to trigger authentication.
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the voice and data VLANs of the port. If used, only one device on the port should enforce per-user ACLs.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms). This allows any type of device to authenticate on the port.

- **802.1x switch supplicant:** You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity. Once the supplicant switch authenticates successfully the port mode changes from access to trunk in an authenticator switch. In a supplicant switch you must manually configure trunk when enabling CISP.
- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

In the default state, when you connect a supplicant switch to an authenticator switch that has BPDU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. Beginning with Cisco IOS Release 15.0(1)SE, you can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** global configuration command temporarily blocks the supplicant port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** global configuration command opens the supplicant port during the authentication period. This is the default behavior.

We strongly recommend using the **dot1x supplicant controlled transient** command on a supplicant switch when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface configuration command.



Note

If you globally enable BPDU guard on the authenticator switch by using the **spanning-tree portfast bpduguard default** global configuration command, entering the **dot1x supplicant controlled transient** command does not prevent the BPDU violation.

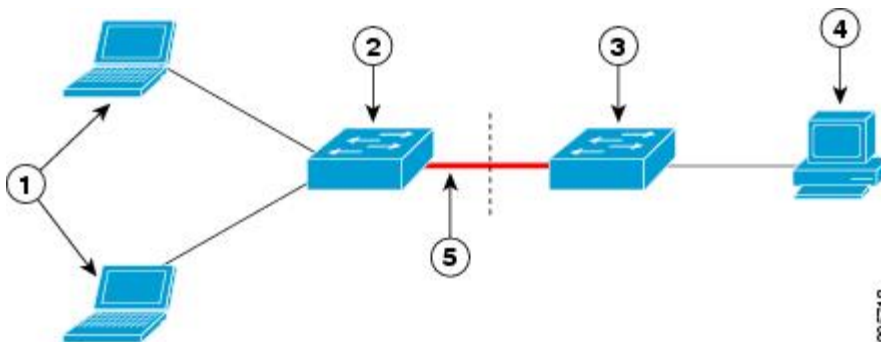
You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

When you reboot an authenticator switch with single-host mode enabled on the interface, the interface may move to err-disabled state before authentication. To recover from err-disabled state, flap the authenticator port to activate the interface again and initiate authentication.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- Host Authorization: Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch.
- Auto enablement: Automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the cisco-av-pair as *device-traffic-class=switch* at the ACS. (You can configure this under the *group* or the *user* settings.)

Figure 25: Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Access control server (ACS)
5	Trunk port		



Note The **switchport nonegotiate** command is not supported on supplicant and authenticator switches with NEAT. This command should not be configured at the supplicant side of the topology. If configured on the authenticator side, the internal macros will automatically remove this command from the port.

Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. In previous releases, when an attempt to authenticate the data client caused a security violation, the entire port shut down, resulting in a complete loss of connectivity.

You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the show commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32 bit integer
- The session start time stamp (a 32 bit integer)

This example shows how the session ID appears in the output of the show authentication command. The session ID in this example is 160000050000000B288508E5:

```
Device# show authentication sessions
Interface  MAC Address  Method  Domain  Status  Session ID
Fa4/0/4    0000.0000.0203  mab     DATA   Authz Success  160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also 160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

How to Configure 802.1x Port-Based Authentication

Default 802.1x Authentication Configuration

Table 35: Default 802.1x Authentication Configuration

Feature	Default Setting
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
AAA	Disabled.

Feature	Default Setting
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Default accounting port • Key 	<ul style="list-style-type: none"> • None specified. • 1645. • 1646. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the dot1x timeout server-timeout interface configuration command.
Inactivity timeout	Disabled.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.

Feature	Default Setting
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.
Voice-aware security	Disabled.

802.1x Authentication Configuration Guidelines

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- You must enable SISF-Based device tracking to use 802.1x authentication. By default, SISF-Based device tracking is disabled on a switch.
- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after re-authentication.

If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.

- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
 - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
 - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
 - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
 - The feature is supported on 802.1x port in single-host mode and multihosts mode.
 - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
 - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.
 - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to re-authenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
 - If the CTS links are in Critical Authentication mode and the active switch reloads, the policy where SGT was configured on a device will not be available on the new active switch. This is because the internal bindings will not be synced to the standby switch in a 3750-X switch stack.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- When wireless guest clients obtains IP from foreign client VLAN instead of anchor client VLAN, you should use the **ip dhcp required** command under the WLAN configuration to force clients to issue a new DHCP request. This prevents the clients from getting an incorrect IP at anchor.
- If the wired guest clients fail to get IP address after a Cisco WLC (foreign) reload, perform a shut/no shut on the ports used by the clients to reconnect them.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines.

- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to re-authorize the port.
- If the port is in the authorized state, the port remains in this state until re-authorization occurs.
- You can configure a timeout period for hosts that are connected by MAC authentication bypass but are inactive. The range is 1 to 65535 seconds.

Maximum Number of Allowed Devices Per Port

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.
- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multihost mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN. An unlimited number of devices are allowed on the voice VLAN.

Configuring 802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **dot1x force-unauthorized**.

Follow these steps to enable the 802.1x readiness check on the switch:

Before you begin

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds

within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated

- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	dot1x test eapol-capable [interface interface-id] Example: Device# dot1x test eapol-capable interface gigabitethernet1/0/13 DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable	Enables the 802.1x readiness check on the switch. (Optional) For <i>interface-id</i> specify the port on which to check for IEEE 802.1x readiness. Note If you omit the optional interface keyword, all interfaces on the switch are tested.
Step 4	dot1x test timeout timeout	(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	startup-config	

Configuring Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature on the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **errdisable detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.



Note If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no shutdown** interface configuration commands.
- You can re-enable individual VLANs by using the **clear errdisable interface interface-id vlan [vlan-list]** privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	errdisable detect cause security-violation shutdown vlan	Shut down any VLAN on which a security violation error occurs. Note If the shutdown vlan keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	errdisable recovery cause security-violation	Enter global configuration mode.

	Command or Action	Purpose
Step 4	clear errdisable interface <i>interface-id</i> vlan <i>[vlan-list]</i>	(Optional) Reenable individual VLANs that have been error disabled. <ul style="list-style-type: none"> • For <i>interface-id</i> specify the port on which to reenable individual VLANs. • (Optional) For <i>vlan-list</i> specify a list of VLANs to be re-enabled. If <i>vlan-list</i> is not specified, all VLANs are re-enabled.
Step 5	Enter the following: <ul style="list-style-type: none"> • shutdown • no shutdown 	(Optional) Re-enable an error-disabled VLAN, and clear all error-disable indications.
Step 6	end	Return to privileged EXEC mode.
Step 7	show errdisable detect	Verify your entries.

Example

This example shows how to configure the switch to shut down any VLAN on which a security violation error occurs:

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

This example shows how to re-enable all VLANs that were error disabled on port Gigabit Ethernet 40/2.

```
Switch# clear errdisable interface gigabitethernet4/0/2  
vlan
```

You can verify your settings by entering the **show errdisable detect** privileged EXEC command.

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- a device connects to an 802.1x-enabled port
- the maximum number of allowed about devices have been authenticated on the port

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables AAA.
Step 3	aaa authentication dot1x {default} method1 Example: Device(config)# <code>aaa authentication dot1x default group radius</code>	Creates an 802.1x authentication method list. To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports. For <i>method1</i> , enter the group radius keywords to use the list of all RADIUS servers for authentication.
Step 4	interface interface-id Example: Device(config)# <code>interface gigabitethernet1/0/4</code>	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 5	switchport mode access Example: Device(config-if)# <code>switchport mode access</code>	Sets the port to access mode.
Step 6	authentication violation {shutdown restrict protect replace} Example: Device(config-if)# <code>authentication violation restrict</code>	Configures the violation mode. The keywords have these meanings: <ul style="list-style-type: none"> • shutdown—Error disable the port. • restrict—Generate a syslog error. • protect—Drop packets from any new device that sends traffic to the port. • replace—Removes the current session and authenticates with the new host.
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if) # end	

Configuring 802.1x Authentication

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

Before you begin

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

Procedure

	Command or Action	Purpose
Step 1	A user connects to a port on the switch.	
Step 2	Authentication is performed.	
Step 3	VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.	
Step 4	The switch sends a start message to an accounting server.	
Step 5	Re-authentication is performed, as necessary.	
Step 6	The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.	
Step 7	The user disconnects from the port.	
Step 8	The switch sends a stop message to the accounting server.	

Configuring 802.1x Port-Based Authentication

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables AAA.
Step 3	aaa authentication dot1x {default} method1 Example: Device(config)# <code>aaa authentication dot1x default group radius</code>	<p>Creates an 802.1x authentication method list.</p> <p>To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports.</p> <p>For <i>method1</i>, enter the group radius keywords to use the list of all RADIUS servers for authentication.</p> <p>Note Though other keywords are visible in the command-line help string, only the group radius keywords are supported.</p>
Step 4	dot1x system-auth-control Example: Device(config)# <code>dot1x system-auth-control</code>	Enables 802.1x authentication globally on the switch.
Step 5	aaa authorization network {default} group radius Example: Device(config)# <code>aaa authorization network default group radius</code>	(Optional) Configures the switch to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment.
Step 6	radius server server name Example: Device(config)# <code>radius server rsim address ipv4 124.2.2.12</code>	(Optional) Specifies the IP address of the RADIUS server.

	Command or Action	Purpose
Step 7	address { ipv4 ipv6 } <i>ip address</i> Example: Device(config-radius-server) # address ipv4 10.0.1.12	Configures the IP address for the RADIUS server.
Step 8	key <i>string</i> Example: Device(config-radius-server) # key rad123	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 9	exit Example: Device(config-radius-server) # exit	Exits the RADIUS server mode and enters the global configuration mode.
Step 10	interface <i>interface-id</i> Example: Device(config) # interface gigabitethernet1/0/2	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 11	switchport mode access Example: Device(config-if) # switchport mode access	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 12	authentication port-control auto Example: Device(config-if) # authentication port-control auto	Enables 802.1x authentication on the port.
Step 13	dot1x pae authenticator Example: Device(config-if) # dot1x pae authenticator	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 14	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device (config-if) # end	

Configuring the Switch-to-RADIUS-Server Communication

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius server** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, the **radius-server retransmit**, and the **key string** global configuration commands.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

Before you begin

You must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius server <i>server name</i> Example: Device (config) # radius server rsim	Specifies the name the RADIUS server and enters radius server configuration mode.
Step 4	address {ipv4 ipv6} ip address auth-port port number acct-port port number Example: Device (config-radius-server) # address ipv4 124.2.2.12	Specifies the IP address of the RADIUS server. For auth-port port-number , specify the UDP destination port for authentication requests. The default is 1645. The range is 0 to 65536.

	Command or Action	Purpose
		For acct-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1646.
Step 5	key string Example: Device(config-radius-server) # key rad123	Specifies the authentication and encryption key used between the Device and the RADIUS daemon running on the RADIUS server. Note The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the radius server command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 6	end Example: Device(config) # end	Returns to privileged EXEC mode.

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **authentication port-control** interface configuration command set to **auto**. Use the **multi-domain** keyword to configure and enable multidomain authentication (MDA), which allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), on the same switch port. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config) # interface	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.

	Command or Action	Purpose
	<code>gigabitethernet2/0/1</code>	
Step 3	<p>authentication host-mode [multi-auth multi-domain multi-host single-host]</p> <p>Example:</p> <pre>Device(config-if)# authentication host-mode multi-host</pre>	<p>Allows multiple hosts (clients) on an 802.1x-authorized port.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • multi-auth—Allow multiple authenticated clients on both the voice VLAN and data VLAN. <p>Note The multi-auth keyword is only available with the authentication host-mode command.</p> <ul style="list-style-type: none"> • multi-host—Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated. • multi-domain—Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an IEEE 802.1x-authorized port. <p>Note You must configure the voice VLAN for the IP phone when the host mode is set to multi-domain.</p> <p>Make sure that the authentication port-control interface configuration command is set to auto for the specified interface.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication periodic Example: Device(config-if)# <code>authentication periodic</code>	Enables periodic re-authentication of the client, which is disabled by default. Note The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the authentication timer reauthenticate command.
Step 4	authentication timer {[inactivity reauthenticate restart unauthorized]} {value}} Example: Device(config-if)# <code>authentication timer reauthenticate 180</code>	Sets the number of seconds between re-authentication attempts. The authentication timer keywords have these meanings: <ul style="list-style-type: none"> • inactivity—Interval in seconds after which if there is no activity from the client then it is unauthorized • reauthenticate—Time in seconds after which an automatic re-authentication attempt is initiated • restart <i>value</i>—Interval in seconds after which an attempt is made to authenticate an unauthorized port • unauthorized <i>value</i>—Interval in seconds after which an unauthorized session will get deleted This command affects the behavior of the switch only if periodic re-authentication is enabled.

	Command or Action	Purpose
Step 5	end Example: Device (config-if) # end	Returns to privileged EXEC mode.

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer restart** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication timer restart <i>seconds</i> Example: Device (config-if) # authentication timer restart 30	Sets the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show authentication sessions interface gigabitethernet2/0/1</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication timer reauthenticate seconds Example: Device(config-if)# <code>authentication timer reauthenticate 60</code>	Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5.

	Command or Action	Purpose
Step 4	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: Device# show authentication sessions interface gigabitethernet2/0/1	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example:	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Device(config)# interface gigabitethernet2/0/1	
Step 3	dot1x max-reauth-req <i>count</i> Example: Device(config-if)# dot1x max-reauth-req 5	Sets the number of times that the switch sends an EAP-request/identity frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device# interface gigabitethernet2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example:	Sets the port to access mode only if you previously configured the RADIUS server.

	Command or Action	Purpose
	Device(config-if)# switchport mode access	
Step 4	dot1x max-req count Example: Device(config-if)# dot1x max-req 4	Sets the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	authentication mac-move permit Example: Device(config)# authentication mac-move permit	<p>Enables MAC move on the switch. Default is deny.</p> <p>In Session Aware Networking mode, the default CLI is access-session mac-move deny. To enable Mac Move in Session Aware Networking, use the no access-session mac-move global configuration command.</p> <p>In legacy mode (IBNS 1.0), default value for mac-move is deny and in C3PL mode (IBNS 2.0) default value is permit.</p>
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 4	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/2</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication violation {protect replace restrict shutdown} Example: Device(config-if)# <code>authentication violation replace</code>	Use the replace keyword to enable MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: <ul style="list-style-type: none"> • protect: the port drops packets with unexpected MAC addresses without generating a system message.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • restrict: violating packets are dropped by the CPU and a system message is generated. • shutdown: the port is error disabled when it receives an unexpected MAC address.
Step 4	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.



Note Periodic AAA accounting updates are not supported. The switch does not send periodic interim accounting records to the accounting server.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```




Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of “Update/Watchdog packets from this AAA client” in your RADIUS server Network Configuration tab. Next, enable “CVS RADIUS Accounting” in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	aaa accounting dot1x default start-stop group radius Example: Device(config-if)# aaa accounting dot1x default start-stop group radius	Enables 802.1x accounting using the list of all RADIUS servers.
Step 4	aaa accounting system default start-stop group radius Example: Device(config-if)# aaa accounting system default start-stop group radius	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/2</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: Device(config-if)# <code>switchport mode private-vlan host</code>	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example:	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094.

	Command or Action	Purpose
	<code>Device(config-if)# authentication event no-response action authorize vlan 2</code>	You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN.
Step 5	end Example: <code>Device(config-if)# end</code>	Returns to privileged EXEC mode.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch stack or a switch, clients that are IEEE 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <code>Device# configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: <code>Device(config)# interface gigabitethernet2/0/2</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: <code>Device(config-if)# switchport mode access</code>	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication port-control auto Example:	Enables 802.1x authentication on the port.

	Command or Action	Purpose
	Device(config-if) # authentication port-control auto	
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Device(config-if) # authentication event fail action authorize vlan 2	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Configuring Number of Authentication Attempts on a Restricted VLAN

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry *retry count*** interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: or	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.

	Command or Action	Purpose
	Device(config-if)# <code>switchport mode access</code>	
Step 4	authentication port-control auto Example: Device(config-if)# <code>authentication port-control auto</code>	Enables 802.1x authentication on the port.
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Device(config-if)# <code>authentication event fail action authorize vlan 8</code>	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	authentication event retry <i>retry count</i> Example: Device(config-if)# <code>authentication event retry 2</code>	Specifies a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	aaa new-model Example: <pre>Device(config)# aaa new-model</pre>	Enables AAA.
Step 3	radius-server dead-criteria {time <i>seconds</i> } [tries <i>number</i>] Example: <pre>Device(config)# radius-server dead-criteria time 20 tries 10</pre>	Sets the conditions that determine when a RADIUS server is considered un-available or down (dead). <ul style="list-style-type: none"> • time— 1 to 120 seconds. The switch dynamically determines a default <i>seconds</i> value between 10 and 60. • number—1 to 100 tries. The switch dynamically determines a default <i>triesnumber</i> between 10 and 100.
Step 4	radius-server deadtime <i>minutes</i> Example: <pre>Device(config)# radius-server deadtime 60</pre>	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 5	radius server <i>server name</i> Example: <pre>Device(config)# radius server rsim address ipv4 124.2.2.12</pre>	(Optional) Specifies the IP address of the RADIUS server.
Step 6	address {ipv4 ipv6} <i>ip address</i> auth-port <i>port_number</i> acct-port <i>port_number</i> Example: <pre>Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560</pre>	Configures the IP address for the RADIUS server.
Step 7	key <i>string</i> Example: <pre>Device(config-radius-server)# key rad123</pre>	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 8	exit Example:	Exits the RADIUS server mode and enters the global configuration mode.

	Command or Action	Purpose
	Device (config-radius-server) # exit	
Step 9	<p>dot1x critical {eapol recovery delay milliseconds}</p> <p>Example:</p> <pre>Device (config) # dot1x critical eapol (config) # dot1x critical recovery delay 2000</pre>	<p>(Optional) Configure the parameters for inaccessible authentication bypass:</p> <ul style="list-style-type: none"> • eapol—Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. • recovery delay milliseconds—Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000 milliseconds (a port can be re-initialized every second).
Step 10	<p>interface interface-id</p> <p>Example:</p> <pre>Device (config) # interface gigabitethernet 1/0/1</pre>	Specify the port to be configured, and enter interface configuration mode.
Step 11	<p>authentication event server dead action {authorize reinitialize} vlan vlan-id]</p> <p>Example:</p> <pre>Device (config-if) # authentication event server dead action reinitialicze vlan 20</pre>	<p>Use these keywords to move hosts on the port if the RADIUS server is unreachable:</p> <ul style="list-style-type: none"> • authorize—Move any new hosts trying to authenticate to the user-specified critical VLAN. • reinitialize—Move all authorized hosts on the port to the user-specified critical VLAN.
Step 12	<p>switchport voice vlan vlan-id</p> <p>Example:</p> <pre>Device (config-if) # switchport voice vlan</pre>	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 13	<p>authentication event server dead action authorize voice</p> <p>Example:</p> <pre>Device (config-if) # authentication event server dead action</pre>	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.

	Command or Action	Purpose
	<code>authorize voice</code>	
Step 14	show authentication interface <i>interface-id</i> Example: <pre>Device(config-if)# do show authentication interface gigabit 1/0/1</pre>	(Optional) Verify your entries.
Step 15	copy running-config startup-config Example: <pre>Device(config-if)# do copy running-config startup-config</pre>	(Optional) Verify your entries.

Example

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius server** global configuration commands. To disable inaccessible authentication bypass, use the **no authentication event server dead action** interface configuration command. To disable critical voice VLAN, use the **no authentication event server dead action authorize voice** interface configuration command.

Example of Configuring Inaccessible Authentication Bypass

This example shows how to configure the inaccessible authentication bypass feature:

```
Device(config)# radius-server dead-criteria time 30 tries 20
Device(config)# radius-server deadtime 60
Device(config)# radius server server1
Device(config-radius-server)# address ipv4 172.29.36.49 acct-port 1618 auth-port 1612
Device(config-radius-server)# key abc1234
Device(config-radius-server)# exit
Device(config)# dot1x critical eapol
Device(config)# dot1x critical recovery delay 2000
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# dot1x critical
Device(config-if)# dot1x critical recovery action reinitialize
Device(config-if)# dot1x critical vlan 20
Device(config-if)# end
```

Configuring 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication control-direction {both in} Example: Device(config-if)# authentication control-direction both	Enables 802.1x authentication with WoL on the port, and use these keywords to configure the port as bidirectional or unidirectional. <ul style="list-style-type: none"> • both—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • in—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: Device# show authentication sessions interface gigabitethernet2/0/3	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 4	mab [eap] Example: Device(config-if)# mab	Enables MAC authentication bypass. (Optional) Use the eap keyword to configure the switch to use EAP for authorization.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring 802.1x User Distribution

Beginning in privileged EXEC mode, follow these steps to configure a VLAN group and to map a VLAN to it:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Device(config)# vlan group eng-dept vlan-list 10	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	no vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Device(config)# no vlan group eng-dept vlan-list 10	Clears the VLAN group configuration or elements of the VLAN group configuration.

Example of Configuring VLAN Groups

This example shows how to configure the VLAN groups, to map the VLANs to the groups, to and verify the VLAN group configurations and mapping to the specified VLANs:

```
Device(config)# vlan group eng-dept vlan-list 10

Device(config)# show vlan group group-name eng-dept
Group Name                Vlans Mapped
-----                -
eng-dept                  10

Device(config)# show dot1x vlan-group all
Group Name                Vlans Mapped
-----                -
eng-dept                  10
hr-dept                   20
```

This example shows how to add a VLAN to an existing VLAN group and to verify that the VLAN was added:

```
Device(config)# vlan group eng-dept vlan-list 30
Device(config)# show vlan group eng-dept
```

```

Group Name                Vlans Mapped
-----
eng-dept                  10,30

```

This example shows how to remove a VLAN from a VLAN group:

```
Device# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
Device(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
```

```
Device(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
Device(config)# no vlan group eng-dept vlan-list all
Device(config)# show vlan-group all
```

For more information about these commands, see the *Cisco IOS Security Command Reference*.

Configuring NAC Layer 2 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example:	Sets the port to access mode only if you configured the RADIUS server.

	Command or Action	Purpose
	Device(config-if)# switchport mode access	
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example: Device(config-if)# authentication event no-response action authorize vlan 8	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 5	authentication periodic Example: Device(config-if)# authentication periodic	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	authentication timer reauthenticate Example: Device(config-if)# authentication timer reauthenticate	Sets re-authentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 7	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 8	show authentication sessions interface <i>interface-id</i> Example: Device# show authentication sessions interface gigabitethernet2/0/3	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring an Authenticator Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.



Note

- The authenticator switch interface configuration must be restored to access mode by explicitly flapping it if a line card is removed and inserted in the chassis when CISP or NEAT session is active.
- The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ISE, which sets the interface as a trunk after the supplicant is successfully authenticated.

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	cisp enable Example: Device(config)# cisp enable	Enables CISP.
Step 3	interface interface-id Example: Device(config)# interface gigabitethernet 2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 4	switchport mode access Example: Device(config-if)# switchport mode access	Sets the port mode to access .
Step 5	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Sets the port-authentication mode to auto.

	Command or Action	Purpose
Step 6	dot1x pae authenticator Example: Device(config-if)# dot1x pae authenticator	Configures the interface as a port access entity (PAE) authenticator.
Step 7	spanning-tree portfast Example: Device(config-if)# spanning-tree portfast trunk	Enables Port Fast on an access port connected to a single workstation or server..
Step 8	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 9	show running-config interface <i>interface-id</i> Example: Device# show running-config interface gigabitethernet 2/0/1	Verifies your configuration.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file. Note Saving changes to the configuration file will mean that the authenticator interface will continue to be in trunk mode after reload. If you want the authenticator interface to remain as an access port, do not save your changes to the configuration file.

Configuring a Supplicant Switch with NEAT

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	cisp enable Example: Device(config)# <code>cisp enable</code>	Enables CISP.
Step 3	dot1x credentials <i>profile</i> Example: Device(config)# <code>dot1x credentials test</code>	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	username <i>suppswitch</i> Example: Device(config)# <code>username suppswitch</code>	Creates a username.
Step 5	password <i>password</i> Example: Device(config)# <code>password myswitch</code>	Creates a password for the new username.
Step 6	dot1x supplicant force-multicast Example: Device(config)# <code>dot1x supplicant force-multicast</code>	Forces the switch to send only multicast EAPOL packets when it receives either unicast or multicast packets. This also allows NEAT to work on the supplicant switch in all host modes.
Step 7	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 8	switchport trunk encapsulation dot1q Example: Device(config-if)# <code>switchport trunk encapsulation dot1q</code>	Sets the port to trunk mode.

	Command or Action	Purpose
Step 9	switchport mode trunk Example: Device(config-if)# switchport mode trunk	Configures the interface as a VLAN trunk port.
Step 10	dot1x pae supplicant Example: Device(config-if)# dot1x pae supplicant	Configures the interface as a port access entity (PAE) supplicant.
Step 11	dot1x credentials <i>profile-name</i> Example: Device(config-if)# dot1x credentials test	Attaches the 802.1x credentials profile to the interface.
Step 12	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 13	show running-config interface <i>interface-id</i> Example: Device# show running-config interface gigabitethernet1/0/1	Verifies your configuration.
Step 14	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.
Step 15	Configuring NEAT with Auto Smartports Macros	You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator switch. For more information, see the <i>Auto Smartports Configuration Guide</i> for this release.

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs



Note You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

Beginning in privileged EXEC mode:

Before you begin

SISF-Based device tracking is a prerequisite to configuring 802.1x authentication. Ensure that you have enabled device tracking programmatically or manually. For more information, see the *Configuring SISF-Based Tracking* chapter.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables AAA.
Step 3	aaa authorization network default local group radius Example: Device(config)# <code>aaa authorization network default local group radius</code>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default local group radius command.
Step 4	radius-server vsa send authentication Example: Device(config)# <code>radius-server vsa send authentication</code>	Configures the radius vsa send authentication.

	Command or Action	Purpose
Step 5	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet2/0/4</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 6	ip access-group <i>acl-id</i> in Example: <pre>Device(config-if)# ip access-group default_acl in</pre>	Configures the default ACL on the port in the input direction. Note The <i>acl-id</i> is an access list name or number.
Step 7	show running-config interface <i>interface-id</i> Example: <pre>Device(config-if)# show running-config interface gigabitethernet2/0/4</pre>	Verifies your configuration.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

Before you begin

SISF-Based device tracking is a prerequisite to configuring 802.1x authentication. Ensure that you have enabled device tracking programmatically or manually.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } { hostname any host } log	Defines the default port ACL.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# access-list 1 deny any log</pre>	<p>The access-list-number is a decimal number from 1 to 99 or 1300 to 1999.</p> <p>Enter deny or permit to specify whether to deny or permit access if conditions are matched.</p> <p>The source is the source address of the network or host that sends a packet, such as this:</p> <ul style="list-style-type: none"> • hostname: The 32-bit quantity in dotted-decimal format. • any: The keyword any as an abbreviation for source and source-wildcard value of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard value. • host: The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. <p>(Optional) Applies the source-wildcard wildcard bits to the source.</p> <p>(Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.</p>
Step 3	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet2/0/2</pre>	Enters interface configuration mode.
Step 4	<p>ip access-group <i>acl-id</i> in</p> <p>Example:</p> <pre>Device(config-if)# ip access-group default_acl in</pre>	<p>Configures the default ACL on the port in the input direction.</p> <p>Note The <i>acl-id</i> is an access list name or number.</p>
Step 5	<p>exit</p> <p>Example:</p> <pre>Device(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	<p>aaa new-model</p> <p>Example:</p> <pre>Device(config)# aaa new-model</pre>	Enables AAA.

	Command or Action	Purpose
Step 7	aaa authorization network default group radius Example: <pre>Device(config)# aaa authorization network default group radius</pre>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default group radius command.
Step 8	radius-server vsa send authentication Example: <pre>Device(config)# radius-server vsa send authentication</pre>	Configures the network access server to recognize and use vendor-specific attributes. Note The downloadable ACL must be operational.
Step 9	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	mab request format attribute 32 vlan access-vlan Example: <pre>Device(config)# mab request format attribute 32 vlan access-vlan</pre>	Enables VLAN ID-based MAC authentication.
Step 3	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Flexible Authentication Ordering

The examples used in the instructions below changes the order of Flexible Authentication Ordering so that MAB is attempted before IEEE 802.1X authentication (dot1x). MAB is configured as the first authentication method, so MAB will have priority over all other authentication methods.

Beginning in privileged EXEC mode, follow these steps:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# switchport mode access	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	authentication order [dot1x mab] {webauth} Example: Device(config-if)# authentication order mab dot1x	(Optional) Sets the order of authentication methods used on a port.
Step 5	authentication priority [dot1x mab] {webauth} Example: Device(config-if)# authentication priority mab dot1x	(Optional) Adds an authentication method to the port-priority list.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring Open1x

Beginning in privileged EXEC mode, follow these steps to enable manual control of the port authorization state:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet 1/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device (config-if)# switchport mode access	Sets the port to access mode only if you configured the RADIUS server.
Step 4	authentication control-direction {both in} Example: Device (config-if)# authentication control-direction both	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	authentication fallback <i>name</i> Example: Device (config-if)# authentication fallback profile1	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
Step 6	authentication host-mode [multi-auth multi-domain multi-host single-host] Example: Device (config-if)# authentication host-mode multi-auth	(Optional) Sets the authorization manager mode on a port.

	Command or Action	Purpose
Step 7	authentication open Example: <pre>Device(config-if)# authentication open</pre>	(Optional) Enables or disable open access on a port.
Step 8	authentication order [dot1x mab] {webauth} Example: <pre>Device(config-if)# authentication order dot1x webauth</pre>	(Optional) Sets the order of authentication methods used on a port.
Step 9	authentication periodic Example: <pre>Device(config-if)# authentication periodic</pre>	(Optional) Enables or disable reauthentication on a port.
Step 10	authentication port-control {auto force-authorized force-un authorized} Example: <pre>Device(config-if)# authentication port-control auto</pre>	(Optional) Enables manual control of the port authorization state.
Step 11	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# <code>switchport mode access</code>	(Optional) Sets the port to access mode only if you configured the RADIUS server.
Step 4	no dot1x pae authenticator Example: Device(config-if)# <code>no dot1x pae authenticator</code>	Disables 802.1x authentication on the port.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface</code>	Enters interface configuration mode, and specify the port to be configured.

	Command or Action	Purpose
	<code>gigabitethernet1/0/2</code>	
Step 3	dot1x default Example: Device(config-if) # <code>dot1x default</code>	Resets the 802.1x parameters to the default values.
Step 4	end Example: Device(config-if) # <code>end</code>	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 36: Privileged EXEC show Commands

Command	Purpose
<code>show dot1x all statistics</code>	Displays 802.1x statistics for all ports
<code>show dot1x interface interface-id statistics</code>	Displays 802.1x statistics for a specific port
<code>show dot1x all [count details statistics summary]</code>	Displays the 802.1x administrative and operational status for a switch
<code>show dot1x interface interface-id</code>	Displays the 802.1x administrative and operational status for a specific port

Table 37: Global Configuration Commands

Command	Purpose
<code>no dot1x logging verbose</code>	Filters verbose 802.1x authentication messages (beginning with Cisco IOS Release 12.2(55)SE)

For detailed information about the fields in these displays, see the command reference for this release.



CHAPTER 19

Configuring Device Sensor

- [About Device Sensor, on page 405](#)
- [MSP-IOS Sensor Device Classifier Interaction, on page 406](#)
- [Configuring Device Sensor, on page 407](#)
- [Configuration Examples for the Device Sensor Feature, on page 412](#)
- [Feature Information for Device Sensor, on page 413](#)

About Device Sensor

Device Sensor uses protocols such as Cisco Discovery Protocol (CDP), Link Layer Discovery Protocol (LLDP), and DHCP to obtain endpoint information from network devices and make this information available to its clients. Device Sensor has internal clients, such as the embedded Device Classifier (local analyzer), Auto Smartports (ASP), MediaNet Service Interface Media Services Proxy, and EnergyWise. Device Sensor also has an external client, Identity Services Engine (ISE), which uses RADIUS accounting to receive and analyze endpoint data. When integrated with ISE, Device Sensor provides central policy management and device-profiling capabilities.



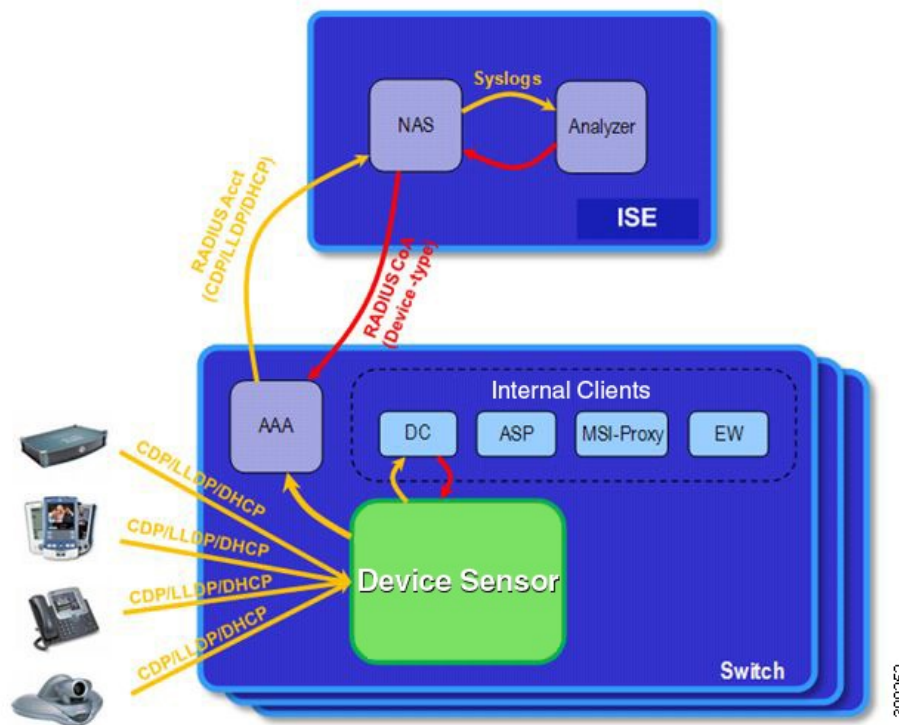
Note Cisco Identity Services Engine (ISE) based profiling is not supported on the LAN Base image.

Device profiling capability consists of two parts:

- Collector--Gathers endpoint data from network devices.
- Analyzer--Processes the data and determines the type of device.

Device Sensor represents the embedded collector functionality. The following illustration shows a Device Sensor in the context of its internal clients and the ISE

Figure 26: Device Sensor Clients



Client notifications and accounting messages that contain profiling data and other session-related data are generated and sent to the internal clients and the ISE. By default, client notifications and accounting events are generated only when an incoming packet includes a Type-Length-Value (TLV) that has not previously been received within a given access session. You can enable client notifications and accounting events for TLV changes; that is, when a previously received TLV is received with a different value.

Device Sensor port security protects a switch from consuming memory and crashing during deliberate or unintentional denial-of-service (DoS)-type attacks. Device Sensor limits the maximum number of device monitoring sessions to 32 per port. While hosts are inactive, the age session limit is 12 hours.

MSP-IOS Sensor Device Classifier Interaction



Note To enable MSP, you must configure the profile flow command. Once done, when SIP, H323, or mDNS traffic are present, appropriate (SIP, H323, or mDNS) TLV notifications are sent to the IOS sensor.

MSP (Media Service Proxy) offers bandwidth reservation for audio or video flows and Metadata services to 3rd-party endpoints. To offer and install Media services, MSP must identify flow attributes and device details. MSP device identification requires automatic identification of various media end points in the network, thereby avoiding any change to the installed end point base. To offer MSP device discovery services, MSP leverages current IOS sensor capability for device classification. (Starting with Release IOS XE 3.3.0SG and IOS 15.1(1)SG, IOS sensor can be used to perform device identification. MSP uses the same functionality with the addition of SIP, H323, and Multicast DNS (mDNS) protocols.) Starting with Release IOS XE 3.4.0SG

and IOS 15.1(2)SG, MSP offers Media services to two kinds of media endpoints: IP Surveillance Cameras and Video-Conferencing Endpoints. Surveillance cameras are identified using mDNS protocol whereas Video-conference-Endpoints are identified using SIP and H.323 protocols.

mDNS compatible devices (Axis, Pelco cameras etc) send mDNS messages for DNS service discovery to a multicast IP address (224.0.0.251) on a standard mDNS port 5353. The mDNS client module listens to this UDP port, receives the mDNS message, and sends it in TLV format to the mDNS IOS sensor shim for further device classification. The module parses the mDNS query and Answer messages fields to create these TLVs.

A Session Initiation Protocol (SIP) registration message is used for SIP based device-discovery and is sent to Cisco Call manager by the SIP Client. A H.225 RAS client registration message is used for H323-based device discovery.

If no Cisco Unified Communicator Manager or GateKeeper exists in the topology, the Endpoint will not generate device Register messages. To handle device discovery in these scenarios, MSP expects the endpoint to make a SIP or H323 call so that MSP snoops the SIP invite or the H323 setup message to identify endpoint details and notify the IOS sensor.

After the IOS sensor receives these protocol details from MSP, the IOS sensor prepares Normalized TLVs, with the new protocols. These protocol details are sent to session manager for further classification.

Configuring Device Sensor

Device Sensor is enabled by default. Complete the following tasks when you want Device Sensor to include or exclude a list of TLVs (termed filter lists) for a particular protocol.



- Note** If you do not perform any Device Sensor configuration tasks, the following TLVs are included by default:
- CDP filter--secondport-status-type and powernet-event-type (types 28 and 29)
 - LLDP filter--organizationally-specific (type 127)
 - DHCP filter--message-type (type 53)

Enabling MSP

You must configure the MSP profile flow command to activate the MSP platform Packet parser. This is because the MSP device handler is tightly coupled with MSP flow parser. Not enabling this command means that MSP will not send SIP, H323 notifications to the IOS sensor.

To enable MSP, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	profile flow Example: Switch(config)# profile flow	Enables MSP. Use the no form of the profile flow command to disable MSP.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Enabling Accounting Augmentation

For the Device Sensor protocol data to be added to accounting messages, you must first enable session accounting by using the following standard Authentication, Authorization, and Accounting (AAA) and RADIUS configuration commands:

```
Switch(config)# aaa new-model
Switch(config)# aaa accounting dot1x default start-stop group radius
Switch(config)# radius-server host{hostname|ip-address}[auth-port port-number][acct-port
port-number] [timeout seconds][retransmit retries][key string]
Switch(config)# radius-server vsa send accounting
```

To add Device Sensor protocol data to accounting records, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor accounting Example: Switch(config)# device-sensor accounting	Enables the addition of sensor protocol data to accounting records and also enables the generation of additional accounting events when new sensor data is detected.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Creating a Cisco Discovery Protocol Filter

To create a CDP filter containing a list of TLVs that can be included or excluded in the Device Sensor output, follow these steps, beginning in privileged EXEC mode:

```
configure terminal
```

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor filter-list cdp list <i>tlv-list-name</i>	

Creating an LLDP Filter

To create an LLDP filter containing a list of TLVs that can be included or excluded in the Device Sensor output, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor filter-list lldp list <i>tlv-list-name</i> Example: Switch(config)# device-sensor filter-list lldp list lldp-list	Creates a TLV list and enters LLDP sensor configuration mode, where you can configure individual TLVs.
Step 3	tlv { name <i>tlv-name</i> number <i>tlv-number</i> } Example: Switch(config-sensor-cdplist)# tlv number 10	Adds individual LLDP TLVs to the TLV list. You can delete the TLV list without individually removing TLVs from the list by using the no device-sensor filter-list lldp list <i>tlv-list-name</i> command.
Step 4	end Example: Switch(config-sensor-llldplist)# end	Returns to privileged EXEC mode.

Creating a DHCP Filter

To create a DHCP filter containing a list of DHCP options that can be included or excluded in the Device Sensor output, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor filter-list dhcp list option-list-name Example: device-sensor filter-list dhcp list option-list-name	Creates an options list and enters DHCP sensor configuration mode, where you can specify individual DHCP options.
Step 3	option { name option-name number option-number } Example: Switch(config-sensor-dhcp-list)# option number 50	Adds individual DHCP options to the option list. You can delete the entire option list without removing options individually from the list by using the no device-sensor filter-list dhcp list option-list-name command.
Step 4	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Applying a Protocol Filter to the Device Sensor Output

Beginning in privileged EXEC mode, follow these steps to apply a CDP, LLDP, or DHCP filter to the sensor output. The output is session notifications to internal sensor clients and accounting requests to the RADIUS server.



Note Only one filter list can be included or excluded at a time.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor filter-spec { cdp dhcp ldp } { exclude { all list list-name } include list list-name } Example: Switch(config)# device-sensor filter-spec cdp include list list1	Applies a specific protocol filter containing a list of protocol TLV fields or DHCP options to the Device Sensor output. <ul style="list-style-type: none"> • cdp –Applies a CDP TLV filter list to the device sensor output

	Command or Action	Purpose
		<ul style="list-style-type: none"> • lldp –Applies an LLDP TLV filter list to the device sensor output. • dhcp –Applies a DHCP option filter list to the device sensor output. • exclude –Specifies the TLVs that must be excluded from the device sensor output. • include –Specifies the TLVs that must be included from the device sensor output. • all –Disables all notifications for the associated protocol. • list list-name –Specifies the protocol TLV filter list name.
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.

Tracking TLV Changes

By default, client notifications and accounting events are generated only when an incoming packet includes a TLV that has not previously been received within a given session.

To enable client notifications and accounting events for TLV changes, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	device-sensor notify all-changes Example: Switch(config)# device-sensor notify all-changes	Enables client notifications and accounting events for all TLV changes, that is, where either a new TLV is received or a previously received TLV is received with a new value in the context of a given session. Note Use the default device-sensor notify or the device-sensor notify new-tlvs command to return to the default TLV.
Step 3	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Switch(config)# end	

Verifying the Device Sensor Configuration

To verify the sensor cache entries for all devices, follow these steps, beginning in privileged EXEC mode:

Procedure

	Command or Action	Purpose
Step 1	show device-sensor cache mac mac-address	Displays sensor cache entries (the list of protocol TLVs or options received from a device) for a specific device. <ul style="list-style-type: none"> • mac-address is the MAC address of the endpoint
Step 2	show device-sensor cache all Example: Switch(config)# device-sensor notify all-changes	Displays sensor cache entries for all devices.

Troubleshooting Commands

The following commands can help troubleshoot Device Sensor.

- `debug device-sensor { errors | events }`
- `debug authentication all`

Restrictions for Device Sensor

- Only CDP, LLDP, and DHCP protocols are supported.
- The session limit for profiling ports is 32.
- The length of one TLV must not be more than 1024 and the total length of TLVs (combined length of TLVs) of all protocols must not be more than 4096.
- Device Sensor profiles devices that are only one hop away.

Configuration Examples for the Device Sensor Feature

The following example shows how to create a CDP filter containing a list of TLVs:

```
Switch> enable
Switch# configure terminal
```

```
Switch(config)# device-sensor filter-list cdp list cdp-list
Switch(config-sensor-cdplist)# tlv name address-type
Switch(config-sensor-cdplist)# tlv name device-name
Switch(config-sensor-cdplist)# tlv number 34
Switch(config-sensor-cdplist)# end
```

The following example shows how to create an LLDP filter containing a list of TLVs:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-list lldp list lldp-list
Switch(config-sensor-lddplist)# tlv name chassis-id
Switch(config-sensor-lddplist)# tlv name management-address
Switch(config-sensor-lddplist)# tlv number 28
Switch(config-sensor-lddplist)# end
```

The following example shows how to create a DHCP filter containing a list of options:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-list dhcp list dhcp-list
Switch(config)# device-sensor filter-list dhcp list dhcp-list
Switch(config-sensor-dhcplist)# option name domain-name
Switch(config-sensor-dhcplist)# option name host-name
Switch(config-sensor-dhcplist)# option number 50
Switch(config-sensor-dhcplist)# end
```

The following example shows how to apply a CDP TLV filter list to the Device Sensor output:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor filter-spec cdp include cdp-list1
```

The following example shows how to enable client notifications and accounting events for all TLV changes:

```
Switch> enable
Switch# configure terminal
Switch(config)# device-sensor notify all-changes
```

Feature Information for Device Sensor

Table 38: Feature Information for Device Sensor

Feature Name	Releases	Feature Information
Device Sensor	Cisco IOS XE 3.6E	This feature was introduced.



CHAPTER 20

Web-Based Authentication

This chapter describes how to configure web-based authentication on the device. It contains these sections:

- [Finding Feature Information, on page 415](#)
- [Web-Based Authentication Overview, on page 415](#)
- [How to Configure Web-Based Authentication, on page 424](#)
- [Monitoring Web-Based Authentication Status, on page 437](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Web-Based Authentication Overview

Use the web-based authentication feature, known as web authentication proxy, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.



Note You can configure web-based authentication on Layer 2 and Layer 3 interfaces.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.



Note HTTPS traffic interception for central web authentication redirect is not supported.



Note You should use global parameter-map (for method-type, custom, and redirect) only for using the same web authentication methods like consent, web consent, and webauth, for all the clients and SSIDs. This ensures that all the clients have the same web-authentication method.

If the requirement is to use Consent for one SSID and Web-authentication for another SSID, then you should use two named parameter-maps. You should configure Consent in first parameter-map and configure webauth in second parameter-map.



Note The traceback that you receive when webauth client tries to do authentication does not have any performance or behavioral impact. It happens rarely when the context for which FFM replied back to EPM for ACL application is already dequeued (possibly due to timer expiry) and the session becomes 'unauthorized'.

Based on where the web pages are hosted, the local web authentication can be categorized as follows:

- *Internal*—The internal default HTML pages (Login, Success, Fail, and Expire) in the controller are used during the local web authentication.
- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the controller and used during the local web authentication.
- *External*—The customized web pages are hosted on the external web server instead of using the in-built or custom web pages.

Based on the various web authentication pages, the types of web authentication are as follows:

- *Webauth*—This is a basic web authentication. Herein, the controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- *Consent or web-passthrough*—Herein, the controller presents a policy page with the Accept or Deny buttons. You need to click the Accept button to access the network.
- *Webconsent*—This is a combination of webauth and consent web authentication types. Herein, the controller presents a policy page with Accept or Deny buttons along with user name or password. You need to enter the correct credentials and click the Accept button to access the network.

Device Roles

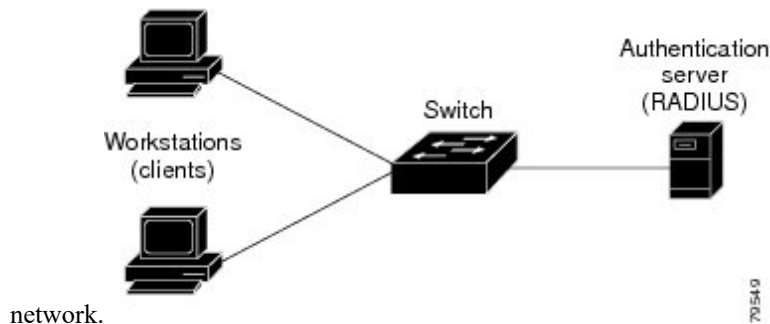
With web-based authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- *Authentication server*—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.

- *Switch*—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 27: Web-Based Authentication Device Roles

This figure shows the roles of these devices in a



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.



Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.

If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.

- Reviews for authorization bypass

If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.

If the server response is access accepted, authorization is bypassed for this host. The session is established.

- Sets up the HTTP intercept ACL

If the server response to the NRH request is access rejected, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.



Note Beginning with Cisco IOS XE Denali 16.1.1 and later, the default session timeout value for web-based authentication on WLC is 1800 seconds. The default session timeout value was infinite seconds, prior to Cisco IOS XE Denali 16.1.1.

- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

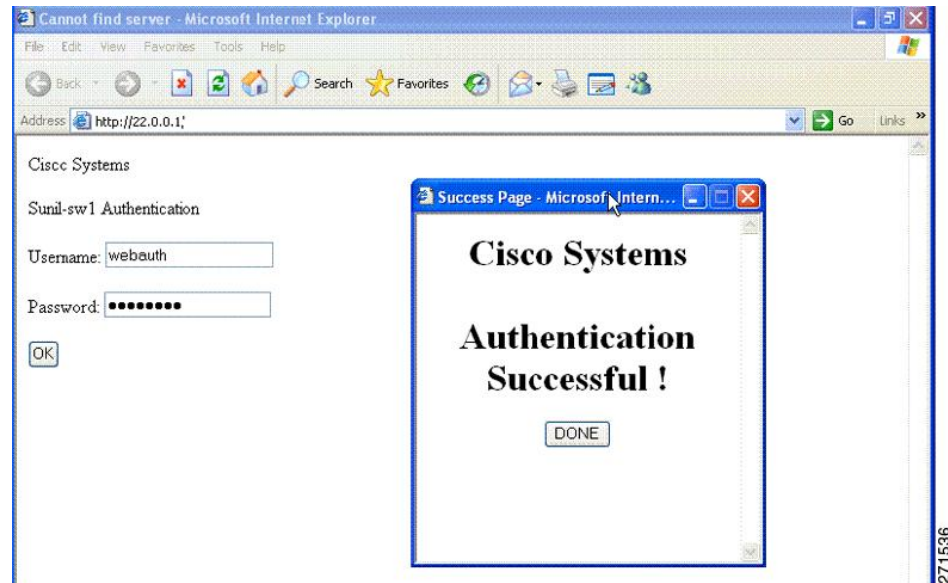
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.
- New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the Login Page. *Cisco Systems* appears on the authentication result pop-up page.

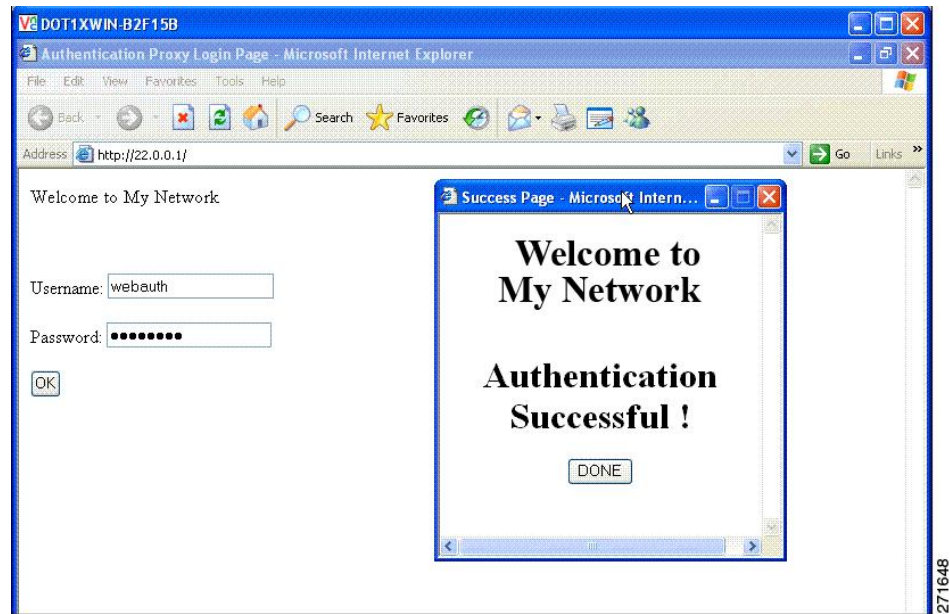
Figure 28: Authentication Successful Banner



The banner can be customized as follows:

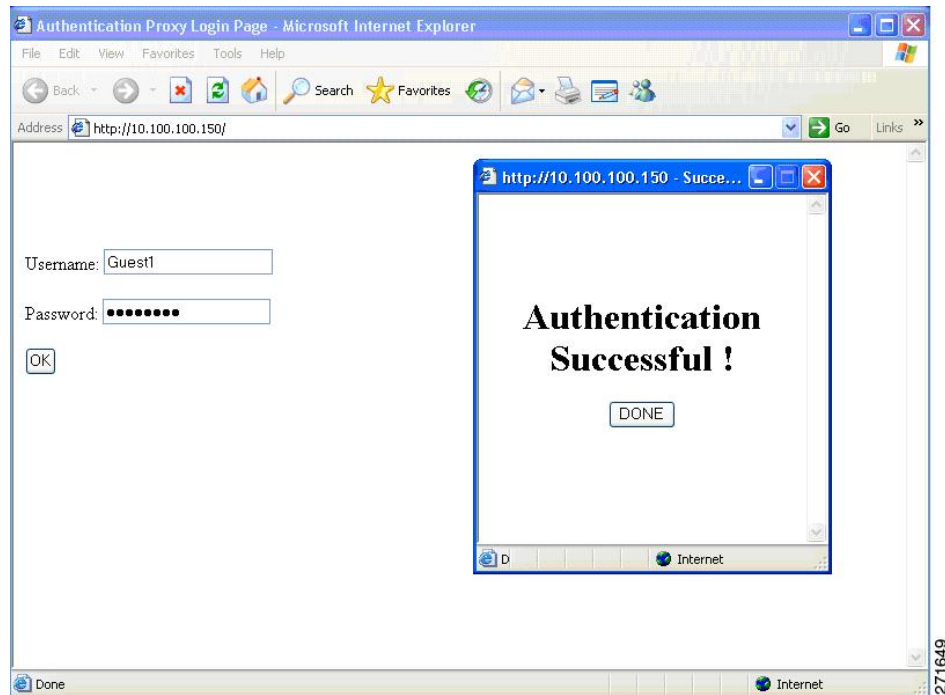
- Add a message, such as switch, router, or company name to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http banner-text** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command
- Add a logo or text file to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http file-path** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command

Figure 29: Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 30: Login Screen With No Banner



For more information, see the *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)* and the *Web Authentication Enhancements - Customizing Authentication Proxy Web Pages*.

Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

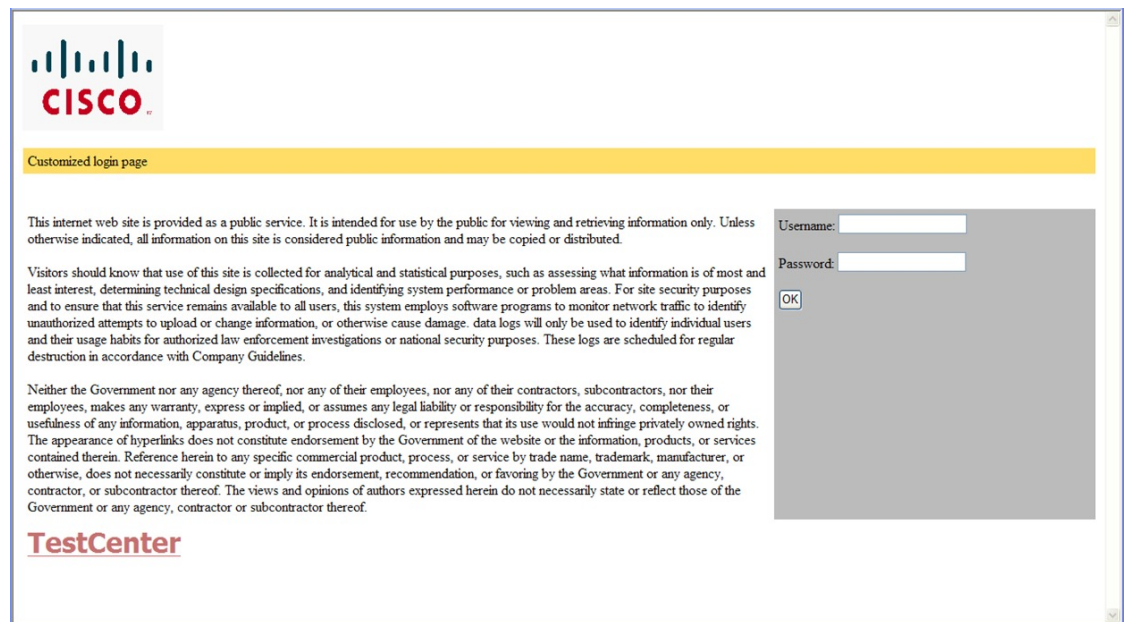
- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, <http://www.cisco.com>). An incomplete URL might cause *page not found* or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- On stackable switches, configured pages can be accessed from the flash on the stack master or members.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the stack master or a member).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 31: Customizable Authentication Page



Authentication Proxy Web Page Guidelines

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.

- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used.
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Web-based Authentication Interactions with Other Features

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, it is more secure, though not required, to configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL. The Policy ACL is applied to the session even if there is no ACL configured on the port.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

How to Configure Web-Based Authentication

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 39: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Key 	<ul style="list-style-type: none"> • None specified • 1645 • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- External web authentication, where the switch redirects a client to a particular host or web server for displaying login message, is not supported.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.

- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must enable SISF-Based device tracking to use web-based authentication. By default, SISF-Based device tracking is disabled on a switch.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.
- Identify the following RADIUS security server settings that will be used while configuring switch-to-RADIUS-server communication:
 - Host name
 - Host IP address
 - Host name and specific UDP port numbers
 - IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

- When you configure the RADIUS server parameters:
 - Specify the **key string** on a separate command line.
 - For **key string**, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.
 - When you specify the **key string**, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.
 - You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server transmit**, and the **radius-server key** global configuration commands.



Note You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DAACL). For more information, see the RADIUS server documentation.

- For a URL redirect ACL:
 - Packets that match a permit access control entry (ACE) rule are sent to the CPU for forwarding to the AAA server.
 - Packets that match a deny ACE rule are forwarded through the switch.
 - Packets that match neither the permit ACE rule or deny ACE rule are processed by the next dACL, and if there is no dACL, the packets hit the implicit-deny ACL and are dropped.

Configuring the Authentication Rule and Interfaces

Follow these steps to configure the authentication rule and interfaces:

Before you begin

SISF-Based device tracking is a prerequisite to Web Authentication. Ensure that you have enabled device tracking programmatically or manually.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip admission name <i>name</i> proxy http Example: Device(config)# ip admission name webauth1 proxy http	Configures an authentication rule for web-based authorization.

	Command or Action	Purpose
Step 4	interface <i>type slot/port</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. <i>type</i> can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 5	ip access-group <i>name</i> Example: Device(config-if)# ip access-group webauthag	Applies the default ACL.
Step 6	ip admission <i>name</i> Example: Device(config)# ip admission name	Configures an authentication rule for web-based authorization for the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show ip admission Example: Device# show ip admission	Displays the configuration.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

If a method-list is configured under VTY lines, the corresponding method list must be added to the AAA configuration:

```
line vty 0 4
  authorization commands 15 abc
aaa authorization commands 15 abc group tacacs+
```

If a method-list is not configured under VTY lines, you must add the default method list to the AAA configuration:

```
line vty 0 4
  aaa authorization commands 15 default group tacacs+
```

Follow these steps to configure AAA authentication:



Note Use default list for AAA authorization, if you are planning to use features such as dACL.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device (config)# aaa new-model	Enables AAA functionality.
Step 4	aaa authentication login default group {tacacs+ radius} Example: Device (config)# aaa authentication login default group tacacs+	Defines the list of authentication methods at login. named_authentication_list refers to any name that is not greater than 31 characters. AAA_group_name refers to the server group name. You need to define the server-group server_name at the beginning itself.
Step 5	aaa authorization auth-proxy default group {tacacs+ radius} Example: Device (config)# aaa authorization auth-proxy default group tacacs+	Creates an authorization method list for web-based authorization.

	Command or Action	Purpose
Step 6	tacacs server <i>server-name</i> Example: Device (config) # tacacs server yourserver	Specifies an AAA server.
Step 7	address { ipv4 ipv6 } <i>ip address</i> Example: Device (config-server-tacacs) # address ipv4 10.0.1.12	Configures the IP address for the TACACS server.
Step 8	key <i>string</i> Example: Device (config-server-tacacs) # key cisco123	Configures the authorization and encryption key used between the switch and the TACACS server.
Step 9	exit Example: Device (config-server-tacacs) # exit	Exits the TACACS server mode and enters the global configuration mode.
Step 10	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 11	show running-config Example: Device# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Switch-to-RADIUS-Server Communication

Follow these steps to configure the RADIUS server parameters:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip radius source-interface vlan <i>vlan interface number</i> Example: Device(config)# ip radius source-interface vlan 80	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 4	radius server <i>server name</i> Example: Device(config)# radius server rsim address ipv4 124.2.2.12	(Optional) Specifies the IP address of the RADIUS server.
Step 5	address {ipv4 ipv6} <i>ip address</i> Example: Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560	Configures the IP address for the RADIUS server.
Step 6	key <i>string</i> Example: Device(config-radius-server)# key rad123	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 7	exit Example:	Exits the RADIUS server mode and enters the global configuration mode.

	Command or Action	Purpose
	<code>Device(config-radius-server)# exit</code>	
Step 8	radius-server dead-criteria tries <i>num-tries</i> Example: <code>Device(config)# radius-server dead-criteria tries 30</code>	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of <i>num-tries</i> is 1 to 100.
Step 9	end Example: <code>Device(config)# end</code>	Returns to privileged EXEC mode.

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the Device. You can enable the server for either HTTP or HTTPS.



Note The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow these steps to enable the server for either HTTP or HTTPS:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <code>Device> enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <code>Device# configure terminal</code>	Enters global configuration mode.
Step 3	ip http server Example:	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.

	Command or Action	Purpose
	Device(config)# <code>ip http server</code>	
Step 4	ip http secure-server Example: Device(config)# <code>ip http secure-server</code>	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login. Note To ensure secure authentication when you enter the ip http secure-server command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the Device default HTML pages during web-based authentication.

For the equivalent Session Aware Networking configuration example for this feature, see the section "Configuring a Parameter Map for Web-Based Authentication" in the chapter, "Configuring Identity Control Policies." of the book, *Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*.

Follow these steps to specify the use of your custom authentication proxy web pages:

Before you begin

Store your custom HTML files on the Device flash memory.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>ip admission proxy http login page file <i>device:login-filename</i></p> <p>Example:</p> <pre>Device(config)# ip admission proxy http login page file disk1:login.htm</pre>	Specifies the location in the Device memory file system of the custom HTML file to use in place of the default login page. The <i>device:</i> is flash memory.
Step 4	<p>ip admission proxy http success page file <i>device:success-filename</i></p> <p>Example:</p> <pre>Device(config)# ip admission proxy http success page file disk1:success.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 5	<p>ip admission proxy http failure page file <i>device:fail-filename</i></p> <p>Example:</p> <pre>Device(config)# ip admission proxy http fail page file disk1:fail.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login failure page.
Step 6	<p>ip admission proxy http login expired page file <i>device:expired-filename</i></p> <p>Example:</p> <pre>Device(config)# ip admission proxy http login expired page file disk1:expired.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Specifying a Redirection URL for Successful Login

Follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip admission proxy http success redirect <i>url-string</i> Example: Device(config)# ip admission proxy http success redirect www.example.com	Specifies a URL for redirection of the user in place of the default login success page.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Web-Based Authentication Parameters

Follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ip admission max-login-attempts <i>number</i> Example: Device(config)# ip admission max-login-attempts 10	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Web-Based Authentication Local Banner

Follow these steps to configure a local banner on a switch that has web authentication configured.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip admission auth-proxy-banner http <i>[banner-text file-path]</i>	Enables the local banner.

	Command or Action	Purpose
	Example: <pre>Device(config)# ip admission auth-proxy-banner http C My Switch C</pre>	(Optional) Create a custom banner by entering <i>C banner-text C</i> (where <i>C</i> is a delimiting character), or <i>file-path</i> that indicates a file (for example, a logo or text file) that appears in the banner.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Follow these steps to remove web-based authentication cache entries:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	clear ip auth-proxy cache <i>{* host ip address}</i> Example: <pre>Device# clear ip auth-proxy cache 192.168.4.5</pre>	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
Step 3	clear ip admission cache <i>{* host ip address}</i> Example:	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a

	Command or Action	Purpose
	Device# <code>clear ip admission cache 192.168.4.5</code>	specific IP address to delete the entry for a single host.

Monitoring Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 40: Privileged EXEC show Commands

Command	Purpose
<code>show authentication sessions method webauth</code>	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
<code>show wireless client mac-address a.a.a detail</code>	Displays the session specific wireless information and wireless states.
<code>show authentication sessions interface type slot/port[details]</code>	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the show access-session interface command.



CHAPTER 21

Configuring Port-Based Traffic Control

- [Overview of Port-Based Traffic Control](#) , on page 439

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported:

- Storm Control
- Protected Ports
- Port Blocking
- Port Security
- Protocol Storm Protection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About Storm Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

How Traffic Activity is Measured

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic
- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in packets per second and for small frames. This feature is enabled globally. The threshold for small frames is configured for each interface.

With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.



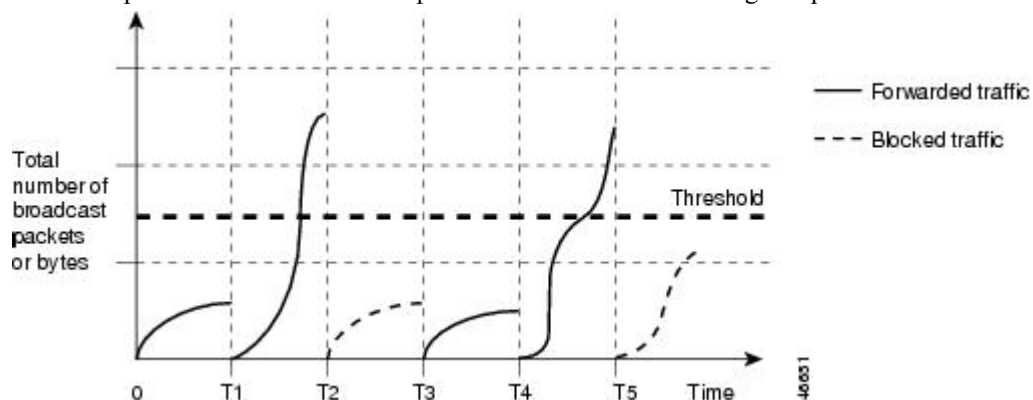
Note

When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

Traffic Patterns

Figure 32: Broadcast Storm Control Example

This example shows broadcast traffic patterns on an interface over a given period of time.



Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2

and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.



Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

How to Configure Storm Control

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.



Note Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Follow these steps to storm control and threshold levels:

Before you begin

Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	storm-control { broadcast multicast unicast } level { <i>level</i> [<i>level-low</i>] bps <i>bps</i> [<i>bps-low</i>] pps <i>pps</i> [<i>pps-low</i>]} Example: Device(config-if)# storm-control unicast level 87 65	Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled. The keywords have these meanings: <ul style="list-style-type: none"> • For <i>level</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. • (Optional) For <i>level-low</i>, specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked. <ul style="list-style-type: none"> • For bps <i>bps</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. • (Optional) For <i>bps-low</i>, specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic

	Command or Action	Purpose
		<p>drops below this level. The range is 0.0 to 10000000000.0.</p> <ul style="list-style-type: none"> For pps pps, specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. (Optional) For pps-low, specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. <p>For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.</p>
Step 5	<p>storm-control action {shutdown trap}</p> <p>Example:</p> <pre>Device(config-if)# storm-control action trap</pre>	<p>Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps.</p> <ul style="list-style-type: none"> Select the shutdown keyword to error-disable the port during a storm. Select the trap keyword to generate an SNMP trap when a storm is detected.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show storm-control [interface-id] [broadcast multicast unicast]</p> <p>Example:</p> <pre>Device# show storm-control gigabitethernet1/0/1 unicast</pre>	<p>Verifies the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, details for all traffic types (broadcast, multicast and unicast) are displayed.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Information About Protected Ports

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

How to Configure Protected Ports

Configuring a Protected Port

Before you begin

Protected ports are not pre-defined. This is the task to configure one.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport protected Example: Device(config-if)# switchport protected	Configures the interface to be a protected port.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Device# show interfaces gigabitethernet1/0/1 switchport	Verifies your entries.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 41: Commands for Displaying Protected Port Settings

Command	Purpose
<code>show interfaces [interface-id] switchport</code>	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Information About Port Blocking

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.



Note With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before you begin

The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport block multicast Example: <pre>Device(config-if)# switchport block multicast</pre>	Blocks unknown multicast forwarding out of the port.
Step 5	switchport block unicast Example: <pre>Device(config-if)# switchport block unicast</pre>	Blocks unknown unicast forwarding out of the port.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show interfaces <i>interface-id</i> switchport Example: <pre>Device# show interfaces gigabitethernet1/0/1 switchport</pre>	Verifies your entries.
Step 8	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 42: Commands for Displaying Port Blocking Settings

Command	Purpose
<code>show interfaces [interface-id] switchport</code>	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Prerequisites for Port Security



Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

Restrictions for Port Security

- The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.
- Port Security is not supported on EtherChannel interfaces.

Information About Port Security

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Types of Secure MAC Addresses

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address *mac-address*** interface configuration command, stored in the address table, and added to the switch running configuration.
- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- Sticky secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

Sticky Secure MAC Addresses

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.
- Running diagnostic tests with port security enabled.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.



Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable

addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

- **shutdown**—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.
- **shutdown vlan**—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 43: Security Violation Mode Actions

Violation Mode	Traffic is forwarded 16	Sends SNMP trap	Sends syslog message	Displays error message 17	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No 18

¹⁶ Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.

¹⁷ The switch returns an error message if you manually configure an address that would cause a security violation.

¹⁸ Shuts down only the VLAN on which the violation occurred.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- **Absolute**—The secure addresses on the port are deleted after the specified aging time.
- **Inactivity**—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Port Security and Switch Stacks

When a switch joins a stack, the new switch will get the configured secure addresses. All dynamic secure addresses are downloaded by the new stack member from the other stack members.

When a switch (either the active switch or a stack member) leaves the stack, the remaining stack members are notified, and the secure MAC addresses configured or learned by that switch are deleted from the secure MAC address table.

Default Port Security Configuration

Table 44: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.
- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.
- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.
- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

This table summarizes port security compatibility with other port-based features.

Table 45: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP ¹⁹ port ²⁰	No
Trunk port	Yes
Dynamic-access port ²¹	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	No
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port ²²	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

¹⁹ DTP=Dynamic Trunking Protocol

²⁰ A port configured with the **switchport mode dynamic** interface configuration command.

²¹ A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

²² You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported:

- Storm Control
- Protected Ports
- Port Blocking
- Port Security
- Protocol Storm Protection

How to Configure Port Security

Enabling and Configuring Port Security

Before you begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport mode {access trunk} Example: Device (config-if)# switchport mode access	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 5	switchport voice vlan <i>vlan-id</i> Example: Device (config-if)# switchport voice vlan 22	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.
Step 6	switchport port-security Example:	Enable port security on the interface.

	Command or Action	Purpose
	<pre>Device(config-if)# switchport port-security</pre>	<p>Note Under certain conditions, when port security is enabled on the member ports in a switch stack, the DHCP and ARP packets would be dropped. To resolve this, configure a shut and no shut on the interface.</p>
Step 7	<p>switchport port-security [maximum value [vlan {vlan-list {access voice}}]]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security maximum 20</pre>	<p>(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.</p> <p>(Optional) vlan—sets a per-VLAN maximum value</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • vlan-list—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
Step 8	<p>switchport port-security violation {protect restrict shutdown shutdown vlan}</p> <p>Example:</p>	<p>(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these:</p>

	Command or Action	Purpose
	<pre>Device(config-if)# switchport port-security violation restrict</pre>	<ul style="list-style-type: none"> <li data-bbox="1073 275 1529 590">• protect—When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. <li data-bbox="1073 611 1529 863"> <p>Note We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.</p> <li data-bbox="1073 894 1529 1188">• restrict—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. <li data-bbox="1073 1199 1529 1367">• shutdown—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. <li data-bbox="1073 1377 1529 1503">• shutdown vlan—Use to set the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs.

	Command or Action	Purpose
		<p>Note When a secure port is in the error-disabled state, you can bring it out of this state by entering the errdisable recovery cause psecure-violation global configuration command. You can manually re-enable it by entering the shutdown and no shutdown interface configuration commands or by using the clear errdisable interface vlan privileged EXEC command.</p>
<p>Step 9</p>	<p>switchport port-security [mac-address mac-address [vlan {vlan-id} {access voice}]]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice</pre>	<p>(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned.</p> <p>Note If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration.</p> <p>(Optional) vlan—sets a per-VLAN maximum value.</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • <i>vlan-id</i>—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN.

	Command or Action	Purpose
		<p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
<p>Step 10</p>	<p>switchport port-security mac-address sticky</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address sticky</pre>	<p>(Optional) Enables sticky learning on the interface.</p>
<p>Step 11</p>	<p>switchport port-security mac-address sticky [<i>mac-address</i> vlan {<i>vlan-id</i> {access voice}}]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice</pre>	<p>(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration.</p> <p>Note If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address.</p> <p>(Optional) vlan—sets a per-VLAN maximum value.</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • <i>vlan-id</i>—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.</p>

	Command or Action	Purpose
Step 12	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 13	show port-security Example: Device# show port-security	Verifies your entries.
Step 14	show running-config Example: Device# show running-config	Verifies your entries.
Step 15	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling and Configuring Port Security Aging

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example:	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	<pre>Device(config)# interface gigabitethernet1/0/1</pre>	
Step 4	<p>switchport port-security aging {static time time type {absolute inactivity}}</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security aging time 120</pre>	<p>Enables or disable static aging for the secure port, or set the aging time or type.</p> <p>Note The switch does not support port security aging of sticky secure addresses.</p> <p>Enter static to enable aging for statically configured secure addresses on this port.</p> <p>For <i>time</i>, specifies the aging time for this port. The valid range is from 0 to 1440 minutes.</p> <p>For type, select one of these keywords:</p> <ul style="list-style-type: none"> • absolute—Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • inactivity—Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show port-security [interface interface-id] [address]</p> <p>Example:</p> <pre>Device# show port-security interface gigabitethernet1/0/1</pre>	Verifies your entries.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Port Security

This table displays port security information.

Table 46: Commands for Displaying Port Security Status and Configuration

Command	Purpose
show port-security [interface <i>interface-id</i>]	Displays port security settings for the switch or for the specified interface, including the maximum allowed number of secure MAC addresses for each interface, the number of secure MAC addresses on the interface, the number of security violations that have occurred, and the violation mode.
show port-security [interface <i>interface-id</i>] address	Displays all secure MAC addresses configured on all switch interfaces or on a specified interface with aging information for each address.
show port-security interface <i>interface-id</i> vlan	Displays the number of secure MAC addresses configured per VLAN on the specified interface.

Configuration Examples for Port Security

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# switchport mode access
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 50
Device(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport mode trunk
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)# switchport access vlan 21
Device(config-if)# switchport mode access
Device(config-if)# switchport voice vlan 22
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 20
Device(config-if)# switchport port-security violation restrict
Device(config-if)# switchport port-security mac-address sticky
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Device(config-if)# switchport port-security mac-address 0000.0000.0003
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Device(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Device(config-if)# switchport port-security maximum 10 vlan access
Device(config-if)# switchport port-security maximum 10 vlan voice
```

Information About Protocol Storm Protection

Protocol Storm Protection

When a switch is flooded with Address Resolution Protocol (ARP) or control packets, high CPU utilization can cause the CPU to overload. These issues can occur:

- Routing protocol can flap because the protocol control packets are not received, and neighboring adjacencies are dropped.
- Spanning Tree Protocol (STP) reconverges because the STP bridge protocol data unit (BPDU) cannot be sent or received.
- CLI is slow or unresponsive.

Using protocol storm protection, you can control the rate at which control packets are sent to the switch by specifying the upper threshold for the packet flow rate. The supported protocols are ARP, ARP snooping, Dynamic Host Configuration Protocol (DHCP) v4, DHCP snooping, Internet Group Management Protocol (IGMP), and IGMP snooping.

When the packet rate exceeds the defined threshold, the switch drops all traffic arriving on the specified virtual port for 30 seconds. The packet rate is measured again, and protocol storm protection is again applied if necessary.

For further protection, you can manually error disable the virtual port, blocking all incoming traffic on the virtual port. You can manually enable the virtual port or set a time interval for automatic re-enabling of the virtual port.



Note Excess packets are dropped on no more than two virtual ports.

Virtual port error disabling is not supported for EtherChannel and Flexlink interfaces

Default Protocol Storm Protection Configuration

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Protocol Storm Protection

Enabling Protocol Storm Protection

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	psp {arp dhcp igmp} pps <i>value</i> Example: Device(config)# psp dhcp pps 35	Configures protocol storm protection for ARP, IGMP, or DHCP. For <i>value</i> , specifies the threshold value for the number of packets per second. If the traffic exceeds this value, protocol storm protection is enforced. The range is from 5 to 50 packets per second.
Step 4	errdisable detect cause psp Example: Device(config)# errdisable detect cause psp	(Optional) Enables error-disable detection for protocol storm protection. If this feature is enabled, the virtual port is error disabled. If this feature is disabled, the port drops excess packets without error disabling the port.
Step 5	errdisable recovery interval <i>time</i> Example: Device	(Optional) Configures an auto-recovery time (in seconds) for error-disabled virtual ports. When a virtual port is error-disabled, the switch auto-recovers after this time. The range is from 30 to 86400 seconds.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 7	show psp config {arp dhcp igmp} Example: Device# show psp config dhcp	Verifies your entries.

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {arp dhcp igmp}	Verify your entries.

Additional References for Port-Based Traffic Control

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 22

Configuring Cisco TrustSec

- [Information about Cisco TrustSec, on page 465](#)
- [Finding Feature Information, on page 465](#)
- [Cisco TrustSec Features, on page 466](#)
- [Feature Information for Cisco TrustSec, on page 468](#)

Information about Cisco TrustSec

Cisco TrustSec provides security improvements to Cisco network devices based on the capability to strongly identify users, hosts, and network devices within a network. TrustSec provides topology-independent and scalable access controls by uniquely classifying data traffic for a particular role. TrustSec ensures data confidentiality and integrity by establishing trust among authenticated peers and encrypting links with those peers.

The key component of Cisco TrustSec is the Cisco Identity Services Engine (ISE). Cisco ISE can provision switches with TrustSec Identities and Security Group ACLs (SGACLs), though these may be configured manually on the switch.

Finding Feature Information

To configure Cisco Trustsec on the switch, see the Cisco TrustSec Switch Configuration Guide at the following URL:

<http://www.cisco.com/c/en/us/td/docs/switches/lan/trustsec/configuration/guide/trustsec.html>

Release notes for Cisco TrustSec General Availability releases are at the following URL:

http://www.cisco.com/en/US/docs/switches/lan/trustsec/release/notes/rn_cts_crossplat.html

For restrictions and limitations on Catalyst 3850 and 3650, see the notes available at the following URL:

http://www.cisco.com/c/en/us/td/docs/switches/lan/trustsec/configuration/guide/trustsec/appa_cat3k.html

Additional information about the Cisco TrustSec solution, including overviews, datasheets, features by platform matrix, and case studies, is available at the following URL:

<http://www.cisco.com/en/US/netsol/ns1051/index.html>

Cisco TrustSec Features

The table below lists the TrustSec features to be eventually implemented on TrustSec-enabled Cisco switches. Successive general availability releases of TrustSec will expand the number of switches supported and the number of TrustSec features supported per switch.

Cisco TrustSec Feature	Description
802.1AE Tagging (MACsec)	<p>Protocol for IEEE 802.1AE-based wire-rate hop-to-hop Layer 2 encryption.</p> <p>Between MACsec-capable devices, packets are encrypted on egress from the transmitting device, decrypted on ingress to the receiving device, and in the clear within the devices.</p> <p>This feature is only available between TrustSec hardware-capable devices.</p> <p>Note This feature is not supported on Catalyst 3850 and Catalyst 3650 switches with Cisco IOS XE Denali 16.1.1</p> <p>Note This feature is not supported on Catalyst 2960x.</p>
Endpoint Admission Control (EAC)	<p>EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth).</p>
Network Device Admission Control (NDAC)	<p>NDAC is an authentication process where each network device in the TrustSec domain can verify the credentials and trustworthiness of its peer device. NDAC utilizes an authentication framework based on IEEE 802.1X port-based authentication and uses EAP-FAST as its EAP method. Successful authentication and authorization in NDAC process results in Security Association Protocol negotiation for IEEE 802.1AE encryption.</p> <p>Note This feature is not supported on Catalyst 2960x.</p>

Cisco TrustSec Feature	Description
Security Group Access Control List (SGACL)	<p>A Security Group Access Control List (SGACL) associates a Security Group Tag with a policy. The policy is enforced upon SGT-tagged traffic egressing the TrustSec domain.</p> <p>Note This feature is not supported on Catalyst 3560cx with 15.2(4)E release.</p>
Cisco TrustSec SGACL High Availability	<p>Cisco TrustSec Security Group access control lists (SGACLs) support the high availability functionality on switches that support the Cisco StackWise technology. Cisco StackWise technology provides stateful redundancy and allows the switch stack to enforce and process access control entries.</p> <p>There is no Cisco TrustSec-specific configuration to enable this functionality.</p> <p>This feature is supported only on Catalyst 3850 and 3650 Series Switches from Cisco IOS XE Release Denali 16.2.1 and higher.</p>
Security Association Protocol (SAP)	<p>After NDAC authentication, the Security Association Protocol (SAP) automatically negotiates keys and the cipher suite for subsequent MACSec link encryption between TrustSec peers. SAP is defined in IEEE 802.11i.</p> <p>Note This feature is not supported on Catalyst 3850 and Catalyst 3650 switches with Cisco IOS XE Denali 16.1.1</p> <p>Note This feature is not supported on Catalyst 2960x.</p>
Security Group Tag (SGT)	<p>An SGT is a 16-bit single label indicating the security classification of a source in the TrustSec domain. It is appended to an Ethernet frame or an IP packet.</p> <p>Note This feature is not supported on Catalyst 2960cx and Catalyst 3560cx.</p>
SGT Exchange Protocol (SXP)	<p>Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.</p>

When both ends of a link support 802.1AE MACsec, SAP negotiation occurs. An EAPOL-key exchange occurs between the supplicant and the authenticator to negotiate a cipher suite, exchange security parameters,

and manage keys. Successful completion of these tasks results in the establishment of a security association (SA).

Depending on your software version and licensing and link hardware support, SAP negotiation can use one of these modes of operation:

- Galois Counter Mode (GCM)—authentication and encryption
- GCM authentication (GMAC)— GCM authentication, no encryption
- No Encapsulation—no encapsulation (clear text)
- Null—encapsulation, no authentication or encryption

Feature Information for Cisco TrustSec

Table 47: Feature Information for Cisco TrustSec

Feature Name	Release	Feature Information
<ul style="list-style-type: none"> • NDAC • SXPv1, SXPv2 • SGT • SGACL Layer2 Enforcement • Interface to SGT and VLAN to SGT mapping. • Subnet to SGT mapping • Layer 3 Port Mapping (PM) • Layer 3 Identity Port Mapping (IPM) • Security Group Name Download • SXP Loop Detection • Policy-based CoA 	Cisco IOS XE 3.3SE	These features were introduced on the Catalyst 3850 and 3650 switches.



CHAPTER 23

Configuring Control Plane Policing

- [Restrictions for CoPP, on page 469](#)
- [Information About CoPP, on page 470](#)
- [How to Configure CoPP, on page 477](#)
- [Configuration Examples for CoPP, on page 481](#)
- [Monitoring CoPP, on page 485](#)
- [Feature History and Information for CoPP, on page 486](#)

Restrictions for CoPP

Restrictions for control plane policing (CoPP) include the following:

- Only ingress CoPP is supported. The **system-cpp-policy** policy-map is available on the control plane interface, and only in the ingress direction.
- Only the **system-cpp-policy** policy-map can be installed on the control plane interface.
- The **system-cpp-policy** policy-map and the 17 system-defined classes cannot be modified or deleted.
- Only the **police** action is allowed under the **system-cpp-policy** policy-map. The police rate for system-defined classes must be configured only in packets per second (pps); for user-defined class maps this must be configured only in bits per second (bps).
- We recommend not disabling the policer for a system-defined class map, that is, do not configure the **no police rate rate pps** command. Doing so affects the overall system health in case of high traffic towards the CPU. Further, even if you disable the policer rate for a system-defined class map, the system automatically reverts to the default policer rate after system bootup in order to protect the system bring-up process.
- One or more CPU queues are part of each class-map. Where multiple CPU queues belong to one class-map, changing the policer rate of a class-map affects all CPU queues that belong to that class-map. Similarly, disabling the policer in a class-map disables all queues that belong to that class-map. See [Table 48: System-Defined Values for CoPP, on page 471](#) for information about which CPU queues belong to each class-map.

Information About CoPP

This chapter describes how control plane policing (CoPP) works on your device and how to configure it.

CoPP Overview

The CoPP feature improves security on your device protecting the CPU from unnecessary traffic and DoS attacks. It can also protect control and management traffic from traffic drops caused by high volumes of other, lower priority traffic.

Your device is typically segmented into three planes of operation, each with its own objective:

- The data plane, to forward data packets.
- The control plane, to route data correctly.
- The management plane, to manage network elements.

You can use CoPP to protect most of the CPU-bound traffic and ensure routing stability, reachability, and packet delivery. Most importantly, you can use CoPP to protect the CPU from a DoS attack.

CoPP uses the modular QoS command-line interface (MQC) and CPU queues to achieve these objectives. Different types of control plane traffic are grouped together based on certain criteria, and assigned to a CPU queue. You can manage these CPU queues by configuring dedicated policers in hardware. For example, you can modify the policer rate for certain CPU queues (traffic-type), or you can disable the policer for a certain type of traffic.

Although the policers are configured in hardware, CoPP does not affect CPU performance or the performance of the data plane. But since it limits the number of packets going to CPU, the CPU load is controlled. This means that services waiting for packets from hardware may see a more controlled rate of incoming packets (the rate being user-configurable).

System-Defined Aspects of CoPP

When you power-up the device for the first time, the system automatically performs the following tasks:

- Looks for policy-map **system-cpp-policy**. If not found, the system creates and installs it on the control-plane.
- Creates 17 class-maps under **system-cpp-policy**.

The next time you power-up the device, the system detects the policy and class maps that have already been created.

- Enables sixteen out of the thirty-two CPU queues (after the policy is installed), with their respective default rate. The CPU queues that are enabled by default and their default rates are indicated in the table *System-Defined Values for CoPP*.

The following table lists the class-maps that the system creates when you load the device. It lists the policer that corresponds to each class-map and one or more CPU queues that are grouped under each class-map. There is a one-to-one mapping of class-map to policer, and a one-to-many mapping of class-map to CPU queues.

Table 48: System-Defined Values for CoPP

Class Maps Names	Policer Index (Policer No.)	CPU queues (Queue No.)	CPU Queues Enabled by Default	Default Policer Rate—in packets per second (pps)
system-cpp-police-data	WK_CPP_POLICE_DATA(0)	WK_CPU_Q_ICMP_GEN(3) WK_CPU_Q_BROADCAST(12)	Yes	200
system-cpp-police-l2-control	WK_CPP_POLICE_L2_CONTROL(1)	WK_CPU_Q_L2_CONTROL(1)	No	500
system-cpp-police-routing-control	WK_CPP_POLICE_ROUTING_CONTROL(2)	WK_CPU_Q_ROUTING_CONTROL(4)	Yes	500
system-cpp-police-control-low-priority	WK_CPP_POLICE_CONTROL_LOW_PRI(3)	WK_CPU_Q_ICMP_REDIRECT(6) WK_CPU_Q_GENERAL_PUNT(25)	No	500
system-cpp-police-punt-webauth	WK_CPP_POLICE_PUNT_WEBAUTH(7)	WK_CPU_Q_PUNT_WEBAUTH(22)	No	1000
system-cpp-police-topology-control	WK_CPP_POLICE_TOPOLOGY_CONTROL(8)	WK_CPU_Q_TOPOLOGY_CONTROL(15)	No	13000
system-cpp-police-multicast	WK_CPP_POLICE_MULTICAST(9)	WK_CPU_Q_TRANSIT_TRAFFIC(18) WK_CPU_Q_MCAST_DATA(30)	Yes	500
system-cpp-police-sys-data	WK_CPP_POLICE_SYS_DATA(10)	WK_CPU_Q_LEARNING_CACHE_OVFL(13) WK_CPU_Q_CRYPTO_CONTROL(23) WK_CPU_Q_EXCEPTION(24) WK_CPU_Q_EGR_EXCEPTION(28) WK_CPU_Q_NFL_SAMPLED_DATA(26) WK_CPU_Q_GOLD_PKT(31) WK_CPU_Q_RPF_FAILED(19)	Yes	100
system-cpp-police-dot1x-auth	WK_CPP_POLICE_DOT1X(11)	WK_CPU_Q_DOT1X_AUTH(0)	No	1000
system-cpp-police-protocol-snooping	WK_CPP_POLICE_PR	WK_CPU_Q_PROTO_SNOOPING(16)	No	500
system-cpp-police-sw-forward	WK_CPP_POLICE_SW_FWD(13)	WK_CPU_Q_SW_FORWARDING_Q(14) WK_CPU_Q_SGT_CACHE_FULL(27) WK_CPU_Q_LOGGING(21)	Yes	1000

Class Maps Names	Policer Index (Policer No.)	CPU queues (Queue No.)	CPU Queues Enabled by Default	Default Policer Rate in packets per second (pps)
system-cpp-police-forus	WK_CPP_POLICE_FORUS(14)	WK_CPU_Q_FORUS_ADDR_RESOLUTION(5) WK_CPU_Q_FORUS_TRAFFIC(2)	No	1000
system-cpp-police-multicast-end-station	WK_CPP_POLICE_MULTICAST_SNOOPING(15)	WK_CPU_Q_MCAST_END_STATION_SERVICE(20)	Yes	2000
system-cpp-default	WK_CPP_POLICE_DEFAULT_POLICER	WK_CPU_Q_DHCP_SNOOPING WK_CPU_Q_SHOW_FORWARD	No	1000

The following table lists the CPU queues and the feature(s) associated with each CPU queue.

Table 49: CPU Queues and Associated Feature(s)

CPU queues (Queue No.)	Feature(s)
WK_CPU_Q_DOT1X_AUTH(0)	IEEE 802.1x Port-Based Authentication
WK_CPU_Q_L2_CONTROL(1)	Dynamic Trunking Protocol (DTP) VLAN Trunking Protocol (VTP) Port Aggregation Protocol (PAgP) Client Information Signaling Protocol (CISP) Message session relay protocol Multiple VLAN Registration Protocol (MVRP) Metropolitan Mobile Network (MMN) Link Level Discovery Protocol (LLDP) UniDirectional Link Detection (UDLD) Link Aggregation Control Protocol (LACP) Cisco Discovery Protocol (CDP) Spanning Tree Protocol (STP)
WK_CPU_Q_FORUS_TRAFFIC(2)	Host such as Telnet, Pingv4 and Pingv6, and SNMP Keepalive / loopback detection Initiate-Internet Key Exchange (IKE) protocol (IPSec)
WK_CPU_Q_ICMP_GEN(3)	ICMP - destination unreachable ICMP-TTL expired

CPU queues (Queue No.)	Feature(s)
WK_CPU_Q_ROUTING_CONTROL(4)	Routing Information Protocol version 1 (RIPv1) RIPv2 Interior Gateway Routing Protocol (IGRP) Border Gateway Protocol (BGP) PIM-UDP Virtual Router Redundancy Protocol (VRRP) Hot Standby Router Protocol version 1 (HSRPv1) HSRPv2 Gateway Load Balancing Protocol (GLBP) Label Distribution Protocol (LDP) Web Cache Communication Protocol (WCCP) Routing Information Protocol next generation (RIPng) Open Shortest Path First (OSPF) Open Shortest Path First version 3(OSPFv3) Enhanced Interior Gateway Routing Protocol (EIGRP) Enhanced Interior Gateway Routing Protocol version 6 (EIGRPv6) DHCPv6 Protocol Independent Multicast (PIM) Protocol Independent Multicast version 6 (PIMv6) Hot Standby Router Protocol next generation (HSRPng) IPv6 control Generic Routing Encapsulation (GRE) keepalive Network Address Translation (NAT) punt Intermediate System-to-Intermediate System (IS-IS)
WK_CPU_Q_FORUS_ADDR_RESOLUTION(5)	Address Resolution Protocol (ARP) IPv6 neighbor advertisement and neighbor solicitation
WK_CPU_Q_ICMP_REDIRECT(6)	Internet Control Message Protocol (ICMP) redirect
WK_CPU_Q_BROADCAST(12)	All types of broadcast
WK_CPU_Q_LEARNING_CACHE_OVFL(13)	Learning cache overflow (Layer 2 + Layer 3)
WK_CPU_Q_SW_FORWARDING_Q(14)	Software forwarding

CPU queues (Queue No.)	Feature(s)
WK_CPU_Q_TOPOLOGY_CONTROL(15)	Spanning Tree Protocol (STP) Resilient Ethernet Protocol (REP) Shared Spanning Tree Protocol (SSTP)
WK_CPU_Q_PROTO_SNOOPING(16)	Address Resolution Protocol (ARP) snooping for Dynamic ARP Inspection (DAI)
WK_CPU_Q_DHCP_SNOOPING(17)	DHCP snooping
WK_CPU_Q_TRANSIT_TRAFFIC(18)	This is used for packets punted by NAT, which need to be handled in the software path.
WK_CPU_Q_RPF_FAILED(19)	Data – mRPF (multicast RPF) failed
WK_CPU_Q_MCAST_END_STATION_SERVICE(20)	Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD) control
WK_CPU_Q_LOGGING(21)	Access control list (ACL) logging
WK_CPU_Q_PUNT_WEBAUTH(22)	Web Authentication
WK_CPU_Q_HIGH_RATE_APP(23)	Broadcast
WK_CPU_Q_CRYPTO_CONTROL(24)	Crypto control
WK_CPU_Q_GENERAL_PUNT(25)	General punt
WK_CPU_Q_NFL_SAMPLED_DATA(26)	Netflow sampled data and Media Services Proxy (MSP)
WK_CPU_Q_SGT_CACHE_FULL(27)	Cache full
WK_CPU_Q_EGR_EXCEPTION(28)	Egress resolution exception
WK_CPU_Q_MCAST_DATA(30)	Data - (S,G) creation Data - local joins Data - PIM Registration Data - SPT switchover Data - Multicast
WK_CPU_Q_GOLD_PKT(31)	Gold

User-Configurable Aspects of CoPP

You can perform these tasks to manage control plane traffic:



Note All `system-cpp-policy` configurations must be saved so they are retained after reboot.

Enable or Disable a Policer for CPU Queues

Enable a policer for a CPU queue, by configuring a policer action (in packets per second) under the corresponding class-map, within the `system-cpp-policy` policy-map.

Disable a policer for CPU queue, by removing the policer action under the corresponding class-map, within the `system-cpp-policy` policy-map.



Note If a default policer is already present, carefully consider and control its removal; otherwise the system may see a CPU hog or other anomalies, such as control packet drops.

Set Policer Rates to Default

Set the policer for CPU queues to their default values, by entering the `cpp system-default` command in global configuration mode.

Create User-Defined Class Maps

If a given traffic class does not have a designated class map, and you want to protect this traffic, you can create specific class maps (with filters) for such traffic packets and add these user-defined class maps to `system-cpp-policy`.

While `system-cpp-policy` is applied in the ingress direction, the forwarding engine driver (FED) changes policers on user-defined class maps to the egress. The filters and the policers in all user-defined classes must therefore be applied as egress classifications and actions, respectively. The policy map itself is unaffected by this change in the direction.

When you add a user-defined class map to `system-cpp-policy`, the system automatically installs it on all 32 CPU queues (in addition to the control plane), resulting in 33 instances of the policy. You can see this by entering the `show platform software fed switch {switch_number} | active | standby} qos policy target` status command in privileged EXEC mode.

The police rate on these class maps is controlled by the Active Queue Management (AQM) policer. AQM provides buffering control of traffic flows prior to queuing a packet into the transmit queue of a port, ensuring that certain flows do not hog the switch packet memory. If the AQM policer feature is enabled, any user-defined police rates exceeding the AQM policer limits are disregarded.

User defined class maps have normal QoS or ACL classification filters.

Upgrading or Downgrading the Software Version

Software Version Upgrades and CoPP

When you upgrade the software version on your device, the system checks and make the necessary updates as required for CoPP (For instance, it checks for the `system-cpp-policy` policy map and creates it if missing). You may also have to complete certain tasks before or after the upgrade activity. This is to ensure that any configuration updates are reflected correctly and CoPP continues to work as expected. Depending on the method you use to upgrade the software, upgrade-related tasks may be optional or recommended in some scenarios, and mandatory in others.

The system actions and user actions for an upgrade, are described here. Also included, are any release-specific caveats.

System Actions for an Upgrade

When you upgrade the software version on your device, the system performs these actions. This applies to all upgrade methods:

- If the device did not have a `system-cpp-policy` policy map before upgrade, then on upgrade, the system creates a default policy map.
- If the device had a `system-cpp-policy` policy map before upgrade, then on upgrade, the system does not re-generate the policy.

User Actions for an Upgrade

User actions for an upgrade – depending on upgrade method:

Upgrade Method	Condition	Action Time and Action	Purpose
Regular ²³	None	After upgrade (required) Enter the <code>cpp system-default</code> command in global configuration mode	To get the latest, default policer rates.

²³ Refers to a software upgrade method that involves a reload of the switch. Can be install or bundle mode.

Software Version Downgrades and CoPP

The system actions and user actions for a downgrade, are described here.

System Actions for a Downgrade

When you downgrade the software version on your device, the system performs these actions. This applies to all downgrade methods:

- The system retains the `system-cpp-policy` policy map on the device, and installs it on the control plane.

User Actions for a Downgrade

User actions for a downgrade:

Upgrade Method	Condition	Action Time and Action	Purpose
Regular ²⁴	None	No action required	Not applicable

²⁴ Refers to a software upgrade method that involves a reload of the switch. Can be install or bundle mode.

If you downgrade the software version and then again upgrade, the system action and user actions that apply are the same as those mentioned for upgrades.

How to Configure CoPP

Enabling a CPU Queue or Changing the Policer Rate

The procedure to enable a CPU queue and change the policer rate of a CPU queue is the same. Follow these steps:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	policy-map <i>policy-map-name</i> Example: Device(config)# policy-map system-cpp-policy Device(config-pmap)#	Enters the policy map configuration mode.
Step 4	class <i>class-name</i> Example: Device(config-pmap)# class system-cpp-police-protocol-snooping Device(config-pmap-c)#	Enters the class action configuration mode. Enter the name of the class that corresponds to the CPU queue you want to enable. See table <i>System-Defined Values for CoPP</i> .
Step 5	police rate <i>rate</i> <i>pps</i> Example: Device(config-pmap-c)# police rate 100 pps Device(config-pmap-c-police)#	Specifies an upper limit on the number of incoming packets processed per second, for the specified traffic class. Note The rate you specify is applied to all CPU queues that belong to the class-map you have specified.
Step 6	exit Example: Device(config-pmap-c-police)# exit	Returns to the global configuration mode.

	Command or Action	Purpose
	Device (config-pmap-c) # exit Device (config-pmap) # exit Device (config) #	
Step 7	control-plane Example: Device (config) # control-plane Device (config-cp) #	Enters the control plane (config-cp) configuration mode
Step 8	service-policy input <i>policy-name</i> Example: Device (config) # control-plane Device (config-cp) # service-policy input system-cpp-policy Device (config-cp) #	Installs system-cpp-policy in FED. This command is required for you to see the FED policy. Not configuring this command will lead to an error.
Step 9	end Example: Device (config-cp) # end	Returns to the privileged EXEC mode.
Step 10	show policy-map control-plane Example: Device# show policy-map control-plane	Displays all the classes configured under <code>system-cpp policy</code> , the rates configured for the various traffic types, and statistics

Disabling a CPU Queue

Follow these steps to disable a CPU queue:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	policy-map <i>policy-map-name</i> Example: Device(config)# policy-map system-cpp-policy Device(config-pmap)#	Enters the policy map configuration mode.
Step 4	class <i>class-name</i> Example: Device(config-pmap)# class system-cpp-police-protocol-snooping Device(config-pmap-c)#	Enters the class action configuration mode. Enter the name of the class that corresponds to the CPU queue you want to disable. See the table, <i>System-Defined Values for CoPP</i> .
Step 5	no police rate <i>rate</i> pps Example: Device(config-pmap-c)# no police rate 100 pps	Disables incoming packet processing for the specified traffic class. Note This disables all CPU queues that belong to the class-map you have specified.
Step 6	end Example: Device(config-pmap-c)# end	Returns to the privileged EXEC mode.
Step 7	show policy-map control-plane Example: Device# show policy-map control-plane	Displays all the classes configured under <i>system-cpp policy</i> and the rates configured for the various traffic types and statistics.

Setting the Default Policer Rates for All CPU Queues

Follow these steps to set the policer rates for all CPU queues to their default rates:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	cpp system-default Example: <pre>Device(config)# cpp system-default Defaulting CPP : Policer rate for all classes will be set to their defaults</pre>	Sets the policer rates for all the classes to the default rate.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to the privileged EXEC mode.
Step 5	show platform hardware fed switch { <i>switch-number</i> <i>active</i> <i>standby</i> } qos que stats internal cpu policer Example: <pre>Device# show platform hardware fed switch 1 qos que stat internal cpu policer</pre>	Displays the rates configured for the various traffic types.

Creating A User-Defined Class Map

Follow these steps to create user-defined class maps in system-cpp-policy and set the policer rates in bps

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	class-map <i>class-map-name</i> Example:	Specify the class map you want to create. Enters the class map configuration mode.

	Command or Action	Purpose
	Device(config)# class-map <i>example_class</i> Device(config-cmap)#	
Step 4	exit Example: Device(config-cmap)# exit Device(config)#	Exits the class map configuration mode.
Step 5	policy-map <i>policy-map-name</i> Example: Device(config)# policy-map system-cpp-policy Device(config-pmap)#	Enter the policy map name. Enters the policy map configuration mode.
Step 6	class-map <i>class-map-name</i> Example: Device(config-pmap)# class <i>example_class</i> Device(config-pmap-c)#	Enters the class action configuration mode. Enter the name of the class.
Step 7	[no] police rate <i>target_bit_rate</i> Example: Device(config-pmap-c)# police 90000	Specifies the bit rate per second, enter a value between 8000 and 10000000000. Note The police rate for user-defined class-maps must not exceed 10000 pps worth of traffic.
Step 8	end Example: Device(config-pmap-c-police)# end Device#	Returns to the privileged EXEC mode.
Step 9	show policy-map control-plane Example: Device# show policy-map control-plane	Displays all the classes configured under <i>system-cpp policy</i> , including the user-defined class maps, and the rates configured.

Configuration Examples for CoPP

Example: Enabling a CPU Queue or Changing the Policer Rate of a CPU Queue

This example shows how to enable a CPU queue or to change the policer rate of a CPU queue. Here the **class system-cpp-police-protocol-snooping** CPU queue is enabled with the policer rate of 2000 pps.

```

Device> enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# police rate 2000 pps
Device(config-pmap-c-police)# end

Device# show policy-map control-plane
Control Plane

Service-policy input: system-cpp-policy

<output truncated>

Class-map: system-cpp-police-dot1x-auth (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: none
  police:
    rate 1000 pps, burst 244 packets
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      drop

Class-map: system-cpp-police-protocol-snooping (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: none
  police:
    rate 2000 pps, burst 488 packets
    conformed 0 bytes; actions:
      transmit
    exceeded 0 bytes; actions:
      drop

<output truncated>

Class-map: class-default (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: any

```

Example: Disabling a CPU Queue

This example shows how to disable a CPU queue. Here the **class system-cpp-police-protocol-snooping** CPU queue is disabled.

```

Device> enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# no police rate 100 pps
Device(config-pmap-c)# end

Device# show running-config | begin system-cpp-policy

```



```

policy-map system-cpp-policy
  class system-cpp-police-data
    police rate 200 pps
  class system-cpp-police-sys-data
    police rate 100 pps
  class system-cpp-police-sw-forward
    police rate 1000 pps
  class system-cpp-police-multicast
    police rate 500 pps
  class system-cpp-police-multicast-end-station
    police rate 2000 pps
  class system-cpp-police-punt-webauth
  class system-cpp-police-l2-control
  class system-cpp-police-routing-control
    police rate 500 pps
  class system-cpp-police-control-low-priority
  class system-cpp-police-wireless-priority1
  class system-cpp-police-wireless-priority2
  class system-cpp-police-wireless-priority3-4-5
  class system-cpp-police-topology-control
  class system-cpp-police-dot1x-auth
  class system-cpp-police-protocol-snooping
  class system-cpp-police-forus
  class system-cpp-default

```

<output truncated>

Example: Setting the Default Policer Rates for All CPU Queues

This example shows how to set the policer rates for all CPU queues to their default and then verify the setting.

```

Device> enable
Device# configure terminal
Device(config)# cpp system-default
Defaulting CPP : Policer rate for all classes will be set to their defaults
Device(config)# end

Device# show platform hardware fed switch 1 qos queue stats internal cpu policer

```

```

(default) (set)
QId PlcIdx Queue Name Enabled Rate Rate Drop
-----
0 11 DOT1X Auth No 1000 1000 0
1 1 L2 Control No 500 400 0
2 14 Forus traffic No 1000 1000 0
3 0 ICMP GEN Yes 200 200 0
4 2 Routing Control Yes 1800 1800 0
5 14 Forus Address resolution No 1000 1000 0
6 3 Punt Copy to ICMP Redirect No 500 400 0
7 6 WLESS PRI-5 No 1000 1000 0
8 4 WLESS PRI-1 No 1000 1000 0
9 5 WLESS PRI-2 No 1000 1000 0
10 6 WLESS PRI-3 No 1000 1000 0
11 6 WLESS PRI-4 No 1000 1000 0
12 0 BROADCAST Yes 200 200 0
13 10 Learning cache ovfl Yes 100 200 0
14 13 Sw forwarding Yes 1000 1000 0
15 8 Topology Control No 13000 13000 0
16 12 Proto Snooping No 500 400 0
17 16 DHCP Snooping No 1000 1000 0
18 9 Transit Traffic Yes 500 400 0

```

Example: Creating a User-Defined Class Map

```

19 10 RPF Failed Yes 100 200 0
20 15 MCAST END STATION Yes 2000 2000 0
21 13 LOGGING Yes 1000 1000 0
22 7 Punt Webauth No 1000 1000 0
23 10 Crypto Control Yes 100 200 0
24 10 Exception Yes 100 200 0
25 3 General Punt No 500 400 0
26 10 NFL SAMPLED DATA Yes 100 200 0
27 2 Low Latency Yes 1800 1800 0
28 10 EGR Exception Yes 100 200 0
29 16 Nif Mgr No 1000 1000 0
30 9 MCAST Data Yes 500 400 0
31 10 Gold Pkt Yes 100 200 0
<output truncated>

```

Example: Creating a User-Defined Class Map

Device

This example shows how to create a user-defined class map, apply it to `system-cpp-policy` and display information about where the policy is applied.

A user-defined class map is applied to `system-cpp-policy`, which means that any control traffic matching the user-defined class map `class-cpp-user` is subject to the aggregate policer, under the user-defined class map. Statistics for the user defined traffic class are reported in Bytes.

```

Device> enable
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# class-map match-any class-cpp-user
Device(config-cmap)# match dscp cs1
Device(config-cmap)# exit
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class class-cpp-user
Device(config-pmap-c)# police rate 2m bps
Device(config-pmap-c-police)# end

Device# show policy-map control-plane
<output truncated>
Class-map: class-cpp-user (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
Match: dscp cs1 (8)
police:
  rate 2000000 bps, burst 62500 bytes
  conformed 0 bytes; actions:
    transmit
  exceeded 0 bytes; actions:
    drop
  conformed 0000 bps, exceeded 0000 bps
<output truncated>

```

When you add a user-defined class map to `system-cpp-policy`, the system automatically installs it on all 32 CPU queues, in addition to the control plane (resulting in 33 instances of the policy).

Note how the direction is display as egress (OUT), even though `system-cpp-policy` is applied in the ingress

```

Device# show platform software fed switch active qos policy target status

TCG status summary:

```

Loc	Interface	IIF-ID	Dir	State: (cfg,opr)	Policy
?:255	Control Plane	0x00000001000001	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-0	0x0000000100000d	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-1	0x0000000100000e	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-2	0x0000000100000f	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-3	0x00000001000010	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-4	0x00000001000011	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-5	0x00000001000012	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-6	0x00000001000013	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-7	0x00000001000014	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-8	0x00000001000015	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-9	0x00000001000016	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-10	0x00000001000017	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-11	0x00000001000018	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-12	0x00000001000019	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-13	0x0000000100001a	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-14	0x0000000100001b	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-15	0x0000000100001c	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-16	0x0000000100001d	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-17	0x0000000100001e	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-18	0x0000000100001f	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-19	0x00000001000020	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-20	0x00000001000021	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-21	0x00000001000022	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-22	0x00000001000023	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-23	0x00000001000024	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-24	0x00000001000025	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-25	0x00000001000026	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-26	0x00000001000027	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-27	0x00000001000028	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-28	0x00000001000029	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-29	0x0000000100002a	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-30	0x0000000100002b	OUT	VALID, SET_INHW	system-cpp-policy
?:0	CoPP-Queue-31	0x0000000100002c	OUT	VALID, SET_INHW	system-cpp-policy

Monitoring CoPP

Use these commands to display policer settings, such as, traffic types and policer rates (user-configured and default rates) for CPU queues:

Command	Purpose
show policy-map control-plane	Displays the rates configured for the various traffic types
show policy-map system-cpp-policy	Displays all the classes configured under system-cpp policy, and policer rates
show platform hardware fed switch { <i>switch-number</i> <i>active</i> <i>standby</i> } qos que stats internal cpu policer	Displays the rates configured for the various traffic types
show platform software fed { <i>switch-number</i> <i>active</i> <i>standby</i> } qos policy target status	Displays information about policy status and the target port type.

Feature History and Information for CoPP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Feature	Release	Feature Information
Control Plane Policing (CoPP) or CPP	Cisco IOS XE 3.2SE	This feature was introduced.
CLI configuration for CoPP	Cisco IOS XE Denali 16.1.2	This feature was made user-configurable. CLI configuration options to enable and disable CPU queues, to change the policer rate, and to set policer rates to default.
User-defined class maps	Cisco IOS XE Everest 16.5.1a	Starting with this release, you can create class maps (with filters) and add these user-defined class maps to system-cpp-policy.