

Cisco Embedded Wireless Controller on Catalyst Access Points Configuration Guide, IOS XE Gibraltar 16.12.x

First Published: 2020-01-07

Last Modified: 2022-02-07

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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

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

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface xxxiii

Document Conventions xxxiii

Related Documentation xxxv

Communications, services, and additional information xxxv

Cisco Bug Search Tool xxxvi

Documentation feedback xxxvi

CHAPTER 1

Cisco Embedded Wireless Controller on Catalyst Access Points 1

New configuration model, tags, and profile 1

Configuration workflow 2

Initial Setup 4

PART I

System Configuration 5

CHAPTER 2

System Configuration 7

Information About New Configuration Model 7

Configure a wireless profile policy (GUI) 9

Configure a wireless policy profile (CLI) 10

Configure a flex profile 11

Configure an AP profile (GUI) 12

Configure an AP profile (CLI) 16

Configure an RF profile (GUI) 17

Configure an RF profile (CLI) 17

Configure policy tag (GUI) 19

Configure a policy tag (CLI) 19

Configure wireless RF tag (GUI) 21

Configure wireless RF tag (CLI) 21
Attach a policy tag and site tag to an AP (GUI) 22
Attach policy tag and site tag to an AP (CLI) 23
AP Filter 24
AP filters 24
Set tag priority (GUI) 25
Set tag priority 25
Create an AP filter (GUI) 26
Create an AP filter (CLI) 26
Configure filter priority (GUI) 27
Configure filter priority (CLI) 28
Verify AP filter configuration 28

CHAPTER 3 **Smart Licensing** 31
Information About Cisco Smart Licensing 31
Creating a Smart Account 33
Using Smart Licensing 34
Using Specified License Reservation (SLR) 34
Enabling Specified License Reservation in CSSM 35
Enabling Smart Software Licensing 36
Registering for Smart License (Connected Mode) 37
Enabling Smart License Reservation 38
Enabling Smart Call Home Reporting 38
Configuring AIR License Level (GUI) 39
Configuring AIR License Level (CLI) 40
Configuring AIR Network Essentials License Level 40
Configuring AIR Network Advantage License Level 41
Verifying Smart Licensing Configurations 41

CHAPTER 4 **Conversion and Migration** 45
Conversion and Migration in Embedded Wireless Controller Capable APs 45
Types of Conversion 45
Access Point Conversion 46
Converting a CAPWAP AP to an Embedded Wireless Controller Capable AP 46

Converting an Embedded Wireless Controller Capable AP to a CAPWAP AP 46
Converting a Single AP to CAPWAP or Embedded Wireless Controller Capable AP (CLI) 46
AP Conversion Deployment Scenarios 47
Network Conversion 49
Converting the Network (CLI) 49
Network Conversion Deployment Scenarios 50
SKU Conversion Scenarios 51
Converting AireOS Mobility Express Network to Embedded Wireless Controller Network 52

PART II **Lightweight Access Points** 53

CHAPTER 5 **Country Codes** 55
Information About Country Codes 55
Prerequisites for Configuring Country Codes 56
Configure country codes (GUI) 56
How to Configure Country Codes 56
Configuration examples for configuring country codes 58
View channel list for country codes 58

CHAPTER 6 **AP Priority** 61
Failover Priority for Access Points 61
Set AP priority (GUI) 61
Set AP priority (CLI) 62

CHAPTER 7 **Rogue per AP** 63
Rogue per AP 63
Enabling Rogue Detection 64
Configure an AP profile (GUI) 64
Configure an AP Profile 67
Define a Wireless Site Tag and Assign an AP Profile (GUI) 69
Define a Wireless Site Tag and Assign an AP Profile (CLI) 69
Associating Wireless Tag to an AP (GUI) 69
Associate Wireless Tag to an AP (CLI) 70

CHAPTER 8 802.11 Parameters for Cisco Access Points 71

2.4-GHz radio support 71

Configure 2.4-GHz radio support for the specified slot number (CLI) 71

5-GHz radio support 73

Configure 5-GHz radio support for the specified slot number (CLI) 73

Dual-band radios in Cisco AP models 75

Limitations 76

Configure default XOR radio support (CLI) 76

Configure XOR radio support for the specified slot number (GUI) 78

Configure XOR radio support for the specified slot number (CLI) 79

Verify XOR radio support 81

Receiver only Dual-Band radio support 81

Receiver only dual-band radio support 81

Configure receiver only Dual-Band parameters for APs 81

Enable CleanAir with receiver Only dual-band radio on a Cisco AP (GUI) 81

Enabling CleanAir with receiver only dual-band radio on a Cisco AP (CLI) 82

Disable receiver only dual-band radio on a Cisco AP (GUI) 82

Disable receiver only dual-band radio on a Cisco AP (CLI) 82

Configure client steering (CLI) 83

Verify Cisco APs with dual-band radios 84

CHAPTER 9 802.1x Support 85

802.1X authentication 85

EAP-FAST protocol 86

EAP-TLS/EAP-PEAP Protocol 86

802.1X authentication limitations 86

Topology 86

Configuring 802.1X authentication type and LSC AP authentication type (GUI) 87

Configure 802.1X authentication type and LSC AP authentication type (CLI) 88

Configure the 802.1X username and password (GUI) 89

Configure the 802.1X username and password (CLI) 89

Enable 802.1X on the switch port (CLI) 90

Verify 802.1X on the switch port 92

Verify the authentication type 92

PART III Radio Resource Management 93

CHAPTER 10 Radio Resource Management 95

Radio resource management 95

Radio resource monitors 96

Transmit power control 96

Override the TPC algorithm with minimum and maximum transmit power settings 97

Dynamic channel assignment 97

Coverage hole detection and correction 99

Restrictions 99

How to Configure RRM 100

Configure neighbor discovery type (CLI) 100

Configuring Transmit Power Control 100

Configure Tx-power control threshold (CLI) 100

Configure the Tx-power level (CLI) 101

Configuring 802.11 RRM Parameters 102

Configure 802.11 channel assignment parameters (CLI) 102

Configure 802.11 coverage hole detection (CLI) 104

Configure 802.11 event logging (CLI) 105

Configure 802.11 statistics monitoring (CLI) 106

Configure the 802.11 performance profile (CLI) 107

Configuring Advanced 802.11 RRM 108

Enable channel assignment (CLI) 108

Restart DCA operation 109

Update power assignment parameters (CLI) 109

Configuring Rogue Access Point Detection in RF Groups 110

Configure rogue AP detection in RF groups (CLI) 110

Monitoring RRM Parameters and RF Group Status 111

Monitor RRM parameters 111

Verify RF group status 112

Examples: RF group configuration 112

Event-Driven Radio Resource Management 113

Configure ED-RRM on the controller (CLI) 113

CHAPTER 11 Coverage Hole Detection 115

Coverage hole detection and correction 115

Configure coverage hole detection (GUI) 115

Configure coverage hole detection (CLI) 116

Configure CHD for RF tag profile (GUI) 117

Configuring CHD for RF profile (CLI) 118

CHAPTER 12 Cisco Flexible Radio Assignment 119

Flexible radio assignments 119

XOR support in 2.4-GHz or 5-GHz bands 120

Configure FRA Radio (CLI) 120

Configure FRA radio (GUI) 122

Flexible radio assignment actions 123

Configure FRA action in default RF profile (CLI) 124

Configure FRA action in 2.4-GHz RF profile (CLI) 125

Verify FRA action configuration 125

CHAPTER 13 XOR Radio Support 127

Dual-band radios in Cisco AP models 127

Limitations 128

Configure default XOR radio support (CLI) 128

Configure XOR radio support for the specified slot number (GUI) 131

Configure XOR radio support for the specified slot number (CLI) 131

Verify XOR radio support 133

CHAPTER 14 Cisco Receiver Start of Packet 135

Receiver start of packet detection threshold 135

Restrictions for Rx SOP 135

Permitted range for the Rx SOP threshold 135

Known behavior 136

Configure Rx SOP (CLI) 136

Customize RF profile (CLI) 137

CHAPTER 15 **Client Limit** 139

Client limits 139

Configure client limit per WLAN (GUI) 140

Configure client limit per WLAN (CLI) 140

CHAPTER 16 **IP Theft** 143

IP theft 143

Lower the idle timeout to prevent false IP theft 143

IP preference levels for IP theft detection 143

Configure IP theft (GUI) 144

Configure IP theft (CLI) 144

Configure the IP theft exclusion timer (CLI) 144

Verify IP theft configuration 145

CHAPTER 17 **Unscheduled Automatic Power Save Delivery** 147

Unscheduled automatic power save delivery overview 147

View unscheduled automatic power save delivery 147

CHAPTER 18 **Enabling USB Port on Access Points** 149

USB port as power source for access points 149

Configure an AP profile (CLI) 150

Configure USB settings for an access point (CLI) 151

USB configuration monitoring for access points (CLI) 152

PART IV **Network Management** 153

CHAPTER 19 **DHCP Option82** 155

DHCP Option 82 155

Configure DHCP Option 82 format 156

Configure DHCP Option 82 global interface 158

Configure DHCP Option 82 globally through server override (CLI) 158

Configure DHCP Option 82 globally through different SVIs (GUI) 158

Configure DHCP Option 82 globally through different SVIs (CLI) 159

Configure DHCP Option 82 through a VLAN interface 160
Configure DHCP Option 82 through option-insert command (CLI) 160
Configure DHCP Option 82 through server-ID-override command (CLI) 161
Configure DHCP Option 82 through a subscriber-ID (CLI) 162
Configure DHCP Option 82 through server-ID-override and subscriber-ID commands (CLI) 163
Configure DHCP Option 82 through different SVIs (CLI) 164
AP DHCP Option 82 support on FlexConnect local switching mode 165
Configure AP DHCP Option82 support 165
Verify AP DHCP Option82 support 166

CHAPTER 20 **RADIUS Realm** 169
RADIUS realms 169
Enable RADIUS realm 170
Configure realm to match the RADIUS server for authentication and accounting 171
Configure AAA policy for WLANs 172
Verify RADIUS-realm configuration 173

CHAPTER 21 **Persistent SSID Broadcast** 177
Persistent SSID broadcast 177
Configure persistent SSID broadcast (CLI) 177
Verify persistent SSID broadcast 178

CHAPTER 22 **Network Monitoring** 179
Network monitoring 179
Status information received synchronously - configuration examples 180
Alarm and Event Information Received Asynchronously - Configuration Examples 181

PART V **System Management** 183

CHAPTER 23 **Network Mobility Services Protocol** 185
Information About Network Mobility Services Protocol 185
Enabling NMSP On-Premises Services 186
Modifying the NMSP Notification Interval for Clients, RFID Tags, and Rogues 186
Modifying the NMSP Notification Threshold for Clients, and Tags 187

Configure NMSP strong cipher (CLI) 187
Verify NMSP settings 188
Examples: NMSP settings configuration 190
Probe RSSI location 190
Configuring Probe RSSI 191
Verifying Probe RSSI 192
RFID tag support 192
Configure RFID tag support (CLI) 193
Verify RFID tag support 193

CHAPTER 24 **Application Visibility and Control** 197
Application visibility and control 197
Guidelines for application visibility and control 197
Prerequisites for application visibility and control 198
Restrictions for application visibility and control 198
Configure AVC 199
Create a flow monitor (CLI) 199
Configure a flow monitor (GUI) 200
Create a flow exporter (CLI) 200
Configure a policy tag (CLI) 201
Attach a policy profile to a WLAN interface (GUI) 202
Attach a policy profile to a WLAN interface (CLI) 202
Attach a policy profile to an AP (CLI) 203
Verify the AVC configuration (CLI) 204
AVC-based selective reanchoring 205
Restrictions for AVC-based selective reanchoring 205
Configure the flow exporter (CLI) 205
Configure the flow monitor (CLI) 206
Configure the AVC reanchoring profile (CLI) 206
Configure the wireless WLAN profile policy (CLI) 207
Verify AVC reanchoring 208
AVC-based selective reanchoring 211
Restrictions for AVC-based selective reanchoring 212
Configure the flow exporter (CLI) 212

Configure the flow monitor (CLI) 212
Configure the AVC reanchoring profile (CLI) 213
Configure the wireless WLAN profile policy (CLI) 214
Verify AVC reanchoring 215

CHAPTER 25 **Cisco Spaces** 219
Configure Cisco Spaces 219
Verify Cisco Spaces configuration 220

CHAPTER 26 **EDCA Parameters** 223
EDCA parameters 223
Configure EDCA parameters (GUI) 223
Configure EDCA parameters (CLI) 224

CHAPTER 27 **802.11 parameters and Band Selection** 225
Band select 225
802.11 bands 226
Restrictions for band selection, 802.11 bands, and parameters 226
Configure band selection (GUI) 227
Configure band selection (CLI) 227
Configure the 802.11 bands (GUI) 228
Configure the 802.11 bands (CLI) 229
Configure a band-select RF profile (GUI) 231
Configure 802.11n parameters (GUI) 232
Configure 802.11n parameters (CLI) 232
Configure 802.11h parameters (CLI) 234
Verify configuration settings using band selection and 802.11 bands commands 234
Example: view the configuration settings for the 5-GHz band 235
Example: view the configuration settings for the 2.4-GHz band 236
Example: view the status of 802.11h parameters 238
Example: verify the band-selection settings 238
Examples: band selection configuration 238
Examples: 802.11 bands configuration 239
Examples: 802.11n configuration 239

Examples: 802.11h configuration 240

CHAPTER 28 **Image Download** 241

Information About Image Download 241

Updates to the AP Image Predownload Status (GUI) 241

Image Download Scenarios 242

Image Download During AP Join 242

Network Software Upgrade (Pre-Download) 243

Methods Supported for Image Download 243

TFTP Image Download Method 244

SFTP Image Download Method 244

Desktop (HTTP) Image Download Method 244

Prerequisites for Image Download 244

Configuring Image Download Profile 245

Configuring TFTP Image Download (GUI) 245

Configuring TFTP Image Download (CLI) 246

Configuring SFTP Image Download (GUI) 247

Configuring SFTP Image Download (CLI) 247

Configuring Desktop (HTTP) Image Download (GUI) 248

Initiating Pre-Download (CLI) 249

Verifying Image Download 250

CHAPTER 29 **Conditional Debug, Radioactive Tracing, and Packet Tracing** 253

Conditional debugging 253

Radioactive tracing 253

Feature history for conditional debug and radioactive tracing 254

Conditional debugging and radioactive tracing 254

Location of tracefiles 254

Configure conditional debugging (GUI) 255

Configure conditional debugging (CLI) 255

Recommended workflow for trace files 257

Copy tracefiles off the box 257

Configuration examples for conditional debugging 258

Verify conditional debugging 258

Example: verify radioactive tracing log for SISF 259
Packet tracing 259
Configure conditional debugging packet tracing (CLI) 260
Configure conditional debugging packet tracing per AP (CLI) 261
Configure conditional debugging packet tracing per client (GUI) 262
Configure conditional debugging packet tracing per client (CLI) 262
Verify conditional debugging packet tracing configuration 263

CHAPTER 30 **Aggressive Client Load Balancing** 265
Aggressive client load balancing 265
Guidelines of aggressive client load balancing 265
Restrictions of aggressive client load balancing 266
Enable aggressive client load balancing (GUI) 266
Configure aggressive client load balancing (GUI) 266
Configure aggressive client load balancing (CLI) 266

CHAPTER 31 **Accounting Identity List** 269
Configure accounting identity list (GUI) 269
Configure accounting identity list (CLI) 269
Configure client accounting (GUI) 270
Configure client accounting (CLI) 270

CHAPTER 32 **Volume metering** 273
Configure volume metering (CLI) 273

CHAPTER 33 **Enabling Syslog Messages in Access Points and Controller for Syslog Server** 275
Syslog messages in access points and controllers 275
Configure the syslog server for the controller (GUI) 275
Configure a syslog server for a controller (CLI) 276
AP logging to the in-memory buffer and flash 278
AP logging to terminal 278
Configure AP logging to a syslog server 278
Configure a syslog server for an AP profile (CLI) 279
Configure the AP syslog settings (GUI) 281

Verify syslog server configurations 281

PART VI Security 285

CHAPTER 34 IPv4 ACLs 287

Network security with ACLs 287

ACL overview 287

Access control entries 288

ACL supported types 288

ACEs and fragmented and unfragmented traffic 288

ACEs and fragmented and unfragmented traffic examples 289

Standard and extended IPv4 ACLs 289

IPv4 ACL switch unsupported features 290

Access list numbers 290

Numbered standard IPv4 ACLs 291

Numbered extended IPv4 ACLs 291

Named IPv4 ACLs 292

ACL logging 293

Hardware and software treatment of IP ACLs 293

IPv4 ACL interface considerations 294

Restrictions for configuring IPv4 access control lists 295

How to configure ACLs 295

Configure IPv4 ACLs (GUI) 295

Configure IPv4 ACLs 296

Create a numbered standard ACL (GUI) 296

Create a numbered standard ACL (CLI) 297

Create a numbered extended ACL (GUI) 298

Create a numbered extended ACL (CLI) 298

Create named standard ACLs (GUI) 301

Create named standard ACLs (CLI) 301

Create extended named ACLs (GUI) 303

Create extended named ACLs (CLI) 303

CHAPTER 35 DNS-Based Access Control Lists 305

DNS-based access control lists 305
Restrictions on DNS-based access control lists 306
Define ACLs 307
Methods for applying ACLs 307
Configure ISE for central web authentication (GUI) 308
View DNS-based access control lists 308
Configuration examples for DNS-based access control lists 309
URL filters 309
FlexConnect mode 310
Define URL filter list (CLI) 310
Apply URL filter list to FlexConnect profile (CLI) 311
FlexConnect client IPv6 support with WebAuth pre and post ACL 312
FlexConnect client IPv6 support with WebAuth pre and post ACL 312
Enable pre-authentication ACL for LWA and EWA (GUI) 313
Enable pre-authentication ACL for LWA and EWA (CLI) 313
Enable post-authentication ACL for LWA and EWA (GUI) 314
Enable post-authentication ACL for LWA and EWA (CLI) 315
Enable DNS ACL for LWA and EWA (GUI) 315
Enable DNS ACL for LWA and EWA (CLI) 316
Verify FlexConnect client IPv6 support with WebAuth pre and post ACL 316

CHAPTER 36 **Allowed List of Specific URLs** 319
Allowed lists of specific URLs 319
Add URL to allowed list 319
Portal resolving to multiple IP addresses 321
Verify URLs on the allowed list 321

CHAPTER 37 **Web-Based Authentication** 323
Local web authentication 323
How local web authentication works 324
Restrictions 325
Roles of devices in local web authentication 326
Banner messages for local web authentication 326
Authentication states in customized local web authentication 328

Best practices for customizing web authentication pages 329
Restrictions for customizing web authentication pages 329
Guidelines for configuring a redirection URL for Successful Login page 330
How to configure local web authentication 330
Configure default local web authentication 330
Configure AAA authentication (GUI) 330
Configure AAA authentication (CLI) 331
Configure the HTTP or HTTPS server (GUI) 332
Configure the HTTP server (CLI) 333
Configure the maximum web authentication request retries 334
Configure a local banner in web authentication page (GUI) 334
Configure a local banner in web authentication page (CLI) 335
Configure trustpoint for local web authentication 336
Configuration examples for local web authentication 336
Example: obtain a web authentication certificate 336
Example: display a web authentication certificate 338
Example: choose the default web authentication login page 338
Example: choosing a CWA login page from an IPv4 external web server 339
Example: choose a customized web authentication login page from an IPv6 external web server 339
Example: assigning login, login failure, and logout pages per WLAN 340
Example: configure preauthentication ACL 340
Example: configure webpassthrough 340
Verify web authentication type 340
External Web Authentication 341
Configure EWA with single WebAuth server address and default ports (80/443) (CLI) 341
Configure EWA with multiple web servers and/or ports different than default (80/443) 343
Configure wired guest EWA with multiple web servers, ports different than default (80/443) 345
Authenticating sleeping clients 346
Guidelines for mobility scenarios 347
Restrictions on authenticating sleeping clients 347
Configure authentication for sleeping clients (GUI) 348
Configure authentication for sleeping clients (CLI) 348

CHAPTER 38 Central Web Authentication 351

Central web authentication 351
How to configure ISE 353
Create an authorization profile 353
Create an authentication rule 354
Create an authorization rule 354
How to configure central web authentication on the controller 355
Configure WLAN (GUI) 355
Configure WLAN (CLI) 356
Configure policy profile (CLI) 358
Configure a policy profile (GUI) 359
Create redirect ACL 360
Configure AAA for CWA 362
Configure redirect ACL in flex profile (GUI) 363
Configure redirect ACL in flex profile (CLI) 363
Troubleshoot CWA 364

CHAPTER 39 **ISE Simplification and Enhancements** 367
Security utilities configuration options 367
Configure multiple RADIUS servers 368
Verify AAA and Radius server configurations 369
Captive bypassing 370
Configure captive bypassing for WLAN in LWA and CWA (GUI) 371
Configure captive bypassing for WLAN in LWA and CWA 371
DHCP option 55 and 77 parameters 372
Configure the DHCP options 55 and 77 to ISE using the GUI 372
Configure the device to send DHCP options 55 and 77 to ISE (CLI) 373
Configure the EAP request timeout using the GUI 374
Configure EAP request timeout 374
Configure the EAP request timeout in wireless security 375
Captive portals 376
Configure captive portal (GUI) 376
Configure captive portal 377
Example: Captive portal configuration 380

CHAPTER 40 **Authentication and Authorization Between Multiple RADIUS Servers** 381

Authentication and authorization between multiple RADIUS servers 381

Configure explicit authentication and authorization server list (GUI) 382

Configure explicit authentication server list (GUI) 383

Configure explicit authentication server list (CLI) 383

Configure explicit authorization server list (GUI) 384

Configure explicit authorization server list (CLI) 385

Configure authentication and authorization list for 802.1X security (GUI) 387

Configure authentication and authorization list for 802.1X security (CLI) 387

Configure web authentication for WLAN with split authentication and authorization servers 388

Configure authentication and authorization list for web authentication (GUI) 388

Configure authentication and authorization list for web authentication (CLI) 389

Verify split authentication and authorization configuration 390

Configuration examples 391

CHAPTER 41 **Secure LDAP** 393

Lightweight Directory Access Protocol (LDAP) 393

SLDAP prerequisite 394

SLDAP restrictions 395

Configure SLDAP (CLI) 395

Configure a AAA server group (GUI) 396

Configure a AAA server group (CLI) 397

Configure search and bind operations for authentication requests (CLI) 398

Configure a dynamic attribute map on a SLDAP server (CLI) 399

Verify the SLDAP configuration 400

CHAPTER 42 **RADIUS DTLS** 401

RADIUS DTLS 401

Requirement: Port RADIUS DTLS support to BINOS platforms 402

Configure RADIUS DTLS server (CLI) 402

Configure RADIUS DTLS connection timeout 403

Configure RADIUS DTLS idle timeout 404

Configure source interface for RADIUS DTLS server 405

Configure RADIUS DTLS port number (CLI) 406
Configure RADIUS DTLS connection retries (CLI) 407
Configure RADIUS DTLS trustpoint (CLI) 408
Configure RADIUS DTLS match-server-identity 409
Configure DTLS Dynamic Author (CLI) 409
Enable DTLS for client (CLI) 410
Configure client trustpoint for DTLS (CLI) 411
Configure DTLS idle timeout (CLI) 412
Configure server trustpoint for DTLS (CLI) 412
Verify RADIUS DTLS server configuration 413
Clear RADIUS DTLS specific statistics 414

CHAPTER 43 **MAC Filtering** 415
MAC Filtering 415
MAC Filtering Configuration Guidelines 415
Configuring MAC Filtering for Local Authentication (CLI) 416
Configuring MAC Filtering (GUI) 418
Configuring MAB for External Authentication (CLI) 418

CHAPTER 44 **Dynamic Frequency Selection** 421
Dynamic frequency selection 421
Configure dynamic frequency selection (GUI) 421
Configure dynamic frequency selection 422
DFS verification 423

CHAPTER 45 **Managing Rogue Devices** 425
Rogue devices and detection 425
Restrictions on rogue detection 426
Rogue AP containment and attack signatures 426
Cisco Prime Infrastructure interaction and rogue detection 426
AP impersonation detection 427
Rogue detection security level 428
Set rogue detection security level (CLI) 430
Configuring rogue detection (GUI) 431

Configure rogue detection (CLI) 431
Configure RSSI deviation notification threshold for rogue APs (CLI) 433
Configure management frame protection (GUI) 433
Configure Management Frame Protection (CLI) 434
Verify management frame protection 435
Verify rogue events 435
Verify rogue detection 436
Examples: rogue detection onfiguration 437
Configure rogue policies (GUI) 438
Configure rogue policies (CLI) 439
Rogue Location Discovery Protocol (RLDP) 440
Guidelines to manage RLDP 440
Best practices for RLDP initiation commands and configuration 441
Restrictions for RLDP 441
Rogue Location Discovery Protocol (RLDP) process 442
Configuring RLDP for Generating Alarms (GUI) 442
Configure RLDP to generate alarms (CLI) 443
Configure a schedule for RLDP (GUI) 443
Configure a schedule for RLDP (CLI) 444
Configure an RLDP for auto-contain (GUI) 445
Configure an RLDP for auto-contain (CLI) 445
Configure RLDP retry times on rogue APs (GUI) 446
Configure RLDP retry times on rogue APs (CLI) 446
Verify rogue AP RLDP 447
Wireless Service Assurance (WSA) rogue events 447
Wireless service assurance rogue events 448
Monitor wireless service assurance rogue events 448
Rogue PMF containment 449
Enable rogue PMF containment 450
Verify PMF containment 451

CHAPTER 46 **Classifying Rogue Access Points** 453
Rogue AP classifications 453
Rogue containment methods 455

Guidelines and restrictions for classifying Rogue APs 455
How to classify rogue APs 456
Classify a rogue AP manually (GUI) 456
Classify rogue APs and clients manually (CLI) 457
Configure rogue classification rules (GUI) 458
Configure rogue classification rules (CLI) 459
Monitor rogue classification rules 461
Examples: classifying rogue AP 461

CHAPTER 47 **Configure Secure Shell** 463
SSH 463
SSH and device access 463
SSH servers and integrated clients 464
SSH configuration guidelines 464
SCP 465
Secure copy protocol 465
SFTP support 466
Prerequisites for configuring SSH 466
Restrictions for configuring SSH 466
Set up the device to run SSH 467
Configure the SSH server 468
Monitor the SSH configuration and status 469

CHAPTER 48 **Private Shared Key** 471
Private preshared keys 471
Limitations 471
How identity PSK authentication works 472
Configure a PSK in a WLAN 473
Configure a PSK in a WLAN using GUI 474
Apply a policy profile to a WLAN (GUI) 474
Apply a policy profile to a WLAN using CLI 475
Verify a private PSK 476

CHAPTER 49 **Multi-Preshared Key** 481

Multi-preshared key 481
Restrictions 483
Procedures 483
Configure a multi-preshared key (GUI) 483
Configure a multi-preshared key (CLI) 486
Verify multi-PSK configurations 487

CHAPTER 50 Multiple Authentications for a Client 491
Multiple authentications for a client 491
Supported combination of authentications for a client 491
Configure multiple authentications for a client 492
Configure a WLAN for 802.1X and LWA 492
Configure a WLAN for 802.1X and LWA (CLI) 493
Configure WLAN for Preshared Key (PSK) and LWA (GUI) 494
Configure WLAN for Preshared Key (PSK) and LWA (CLI) 495
Configure WLAN for PSK or iPSK and CWA 496
Configure WLAN for PSK or iPSK and CWA (GUI) 496
Configure WLAN for PSK or iPSK and CWA (CLI) 497
Apply a policy profile to a WLAN 498
Verify multiple authentication configurations 499

CHAPTER 51 Locally Significant Certificates 503
Locally significant certificates 503
Certificate provisioning in controllers 504
How device certificate enrollment works 504
How certificate provisioning works on lightweight APs 505
Restrictions for locally significant certificates 506
Provisioning locally significant certificates 506
Configure RSA key for PKI trustpoint 506
Configure PKI trustpoint parameters 507
Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI) 508
Authenticate and enroll the PKI trustpoint with CA server (CLI) 509
Configure AP join attempts with LSC certificate 511
Configure AP join attempts with LSC certificate (CLI) 511

Configure subject-name parameters in LSC certificate 512
Configure key size for LSC certificate 513
Configure trustpoint for LSC provisioning on an AP 513
Configure an AP LSC provision list (GUI) 514
Configure an AP LSC provision list (CLI) 514
Configure LSC provisioning for all the APs (GUI) 515
Configure LSC provisioning for all APs (CLI) 516
Configure LSC provisioning for the APs in the provision list 517
Verify LSC configuration 517
Configure management trustpoint to LSC 518
Configure management trustpoint to LSC (CLI) 518
AP certificate renewal (LSC) 519
AP certificate renewal (LSC) 519
Benefits of AP certificate renewal (LSC) 520
Requirements for AP certificate renewal (LSC) 520
Restrictions for AP certificate renewal (LSC) 520
Configure AP certificate auto-renewal (LSC) (GUI) 520
Configure AP certificate renewal (LSC) (CLI) 521
Configure forced renewal of AP certificate (LSC) for individual or multiple APs (GUI) 523
Configure forced renewal of AP Certificate (LSC) based on site tag (GUI) 523
Configure forced renewal of AP Certificate (LSC) based on site tag (CLI) 523
Configure forced renewal of all APs (GUI) 524
Verify AP certificate renewal (LSC) 524
MIC and LSC APs joining the controller 526
Support for MIC and LSC APs joining the controller 526
Recommendations and limitations 526
MIC and LSC access point authentication 526
Configure LSC on the controller (CLI) 527
Enable the AP certificate policy on the APs (CLI) 528
Configure the AP policy certificate (GUI) 529
Configure the allowed list of APs to join the controller (CLI) 529
Verifying the configuration status 530
LSC fallback APs 531
LSC fallback APs 531

Troubleshoot LSC fallback state 531
Recover from LSC fallback 532
Configuring controller self-signed certificate for wireless AP join 533
Use cases for controller self-signed certificate for wireless AP join 533
Prerequisites for controller self-signed certificate for wireless AP join 534
Configure clock calendar (CLI) 534
Enable HTTP server using CLI 535
Configure CA server (CLI) 535
Configure trustpoint (CLI) 537
Authenticate and enroll the PKI trustpoint with CA server 539
Tag wireless management trustpoint name 540
Disable PKI server 540
Verify controller certificates for wireless AP join 541

PART VII **Quality of Service** 543

CHAPTER 52 **Quality of Service** 545
Wireless Quality of Service 545
Wireless QoS targets 546
Service set identifiers policies 546
Client policies 546
Supported QoS features on wireless targets 546
Precious metal policies for wireless QoS 547
Prerequisites for wireless QoS 548
Restrictions for QoS on wireless targets 548
Restriction: Client connectivity and TCP packet drops on SSIDs with AAA rate limiting 549
Metal Policy Format 550
Metal policy format 550
Auto QoS policy format 553
Architecture for voice, video and integrated data (AVVID) 558
How to apply Bi-Directional Rate Limiting 560
Bidirectional rate limiting 560
Requirements for bidirectional rate limiting configuration 561
Configure the metal policy on the SSID 561

Configure a metal policy on your client device 562
Configure bidirectional rate limiting for all traffic 562
Configure Bi-Directional Rate Limiting Based on Traffic Classification 563
Apply bidirectional rate limiting policy map to policy profile 565
Apply metal policy with bidirectional rate limiting 565
How to apply Per Client Bi-Directional Rate Limiting 567
Per-client bidirectional rate limiting 567
Prerequisites for per-client bidirectional rate limiting 567
Restrictions on per-client bidirectional rate limiting 568
Configure per-client bidirectional rate limiting (GUI) 568
Verify per client bi-directional rate limiting 568
Configure bi-directional rate limiting using AAA override (CLI) 569
Verify bi-directional rate-limit 569
How to Configure Wireless QoS 571
Configure a policy map with class map (GUI) 571
Configure a class map (CLI) 572
Configuring Policy Profile to Apply QoS Policy (GUI) 573
Configure policy profile to apply QoS policy (CLI) 573
Apply policy profile to policy tag (GUI) 574
Apply policy profile to policy tag (CLI) 574
Attach policy tag to an AP 575
Configure custom QoS mapping (CLI) 576
Configure a DSCP-to-user priority mapping exception 577
Configure trust upstream DSCP value 579

CHAPTER 53 **Wireless Auto-QoS** 581
Auto QoS information 581
Configure wireless auto-QoS on profile policy (CLI) 582
Disable wireless autoqos 582
Rollback auto-QoS configuration (GUI) 583
Rollback AutoQoS configuration (CLI) 584
Clear wireless autoqos policy profile (GUI) 584
Clear wireless autoqos policy profile (CLI) 585
View Auto-QoS on policy profile (CLI) 585

CHAPTER 54 **Native Profiling** 587

Native profiling 587

Create a class map (GUI) 588

Create a class map (CLI) 589

Create a service template (GUI) 590

Create a service template (CLI) 591

Create a parameter map (CLI) 592

Create a policy map (GUI) 592

Create a policy map (CLI) 593

Configure native profiling in local mode (CLI) 595

Verify native profile configuration 595

PART VIII **CleanAir** 597

CHAPTER 55 **Cisco CleanAir** 599

Cisco ISE systems 599

Cisco ISE-related terms 600

Cisco CleanAir components 601

Prerequisites for CleanAir 601

Restrictions for CleanAir 602

Difference between Wi-Fi chip-based RF Management and Cisco CleanAir 602

Persistence interference 603

Spontaneous interference 604

EDRRM and AQR update mode 604

Restrictions for EDRRM and AQR update mode 605

Configure CleanAir 605

Enable CleanAir for the 2.4-GHz band (GUI) 605

Enable CleanAir for the 2.4-GHz band (CLI) 605

Configure a CleanAir alarm for 2.4-GHz air-quality and devices (CLI) 606

Configure interference reporting for a 2.4-GHz device (GUI) 607

Configure interference reporting for a 2.4-GHz device (CLI) 607

Enable CleanAir for the 5-GHz band (GUI) 608

Enable CleanAir for the 5-GHz band (CLI) 609

Configure interference reporting for a 5-GHz device (GUI) 609
Configure interference reporting for a 5-GHz device (CLI) 610
Configure event driven RRM for a CleanAir event (GUI) 611
Configure EDRRM for a CleanAir event (CLI) 612
Configuration examples for CleanAir 613
Verify CleanAir parameters 613
Configure air quality traps (CLI) 614
Monitor with CleanAir device clusters 615
CleanAir FAQs 615

CHAPTER 56 Spectrum Intelligence 617
Spectrum Intelligence 617
Configure spectrum intelligence (CLI) 618
Verify spectrum intelligence information 618

PART IX WLAN 621

CHAPTER 57 WLANs 623
Information About WLANs 623
Band selection 623
Off-Channel Scanning Deferral 623
DTIM period 624
Session Timeouts 625
Cisco Client Extensions 625
Peer-to-peer blocking 625
Diagnostic channels 626
Prerequisites for WLANs 626
Restrictions for WLANs 626
How to Configure WLANs 628
Create WLANs (GUI) 628
Create WLANs (CLI) 628
Delete WLANs (GUI) 629
Delete WLANs (CLI) 629
WLAN search methods (CLI) 630

Enable WLANs (GUI) 631
Enable WLANs (CLI) 631
Disable WLANs (GUI) 632
Disable WLANs (CLI) 632
Configure general WLAN properties (CLI) 633
Configure advanced WLAN properties (CLI) 634
Configure advanced WLAN properties (GUI) 636
WLAN properties verification (CLI) 637

CHAPTER 58 **Network Access Server Identifier** 639
Network Access Server Identifier (NAS-ID) 639
Create a NAS ID policy (GUI) 640
Create a NAS ID policy (CLI) 640
Attach a policy to a tag (GUI) 641
Attach a policy to a tag (CLI) 641
Verify the NAS ID configuration 642

CHAPTER 59 **DHCP for WLANs** 645
DHCP for WLANs 645

CHAPTER 60 **WLAN Security** 647
AAA override 647
Prerequisites for Layer 2 Security 647
How to Configure WLAN Security 648
Configure static WEP layer 2 security parameters (CLI) 648
Configure WPA + WPA2 layer 2 security parameters (CLI) 648

CHAPTER 61 **Peer-to-Peer Client Support** 651
Peer-to-peer client support 651
Configure peer-to-peer client support 651

CHAPTER 62 **802.11r BSS Fast Transition** 653
802.11r Fast Transition 653

Restrictions for 802.11r Fast Transition 654
Monitor 802.11r fast transition 655
Configure 802.11r Fast Transition in an open WLAN (CLI) 656
Configure 802.11r BSS Fast Transition on a Dot1x security enabled WLAN (CLI) 658
Configure 802.11r Fast Transition on a PSK security–enabled WLAN (CLI) 659
Disable 802.11r fast transition (GUI) 660
Disable 802.11r Fast Transition (CLI) 660

CHAPTER 63 **Assisted Roaming** 663
802.11k neighbor list and assisted roaming 663
Restrictions for assisted roaming 664
How to Configure Assisted Roaming 664
Configure assisted roaming (CLI) 664
Verify assisted roaming 666
Configuration examples for assisted roaming 666

CHAPTER 64 **802.11v** 667
802.11v amendment 667
Prerequisites for configuring 802.11v 668
Restrictions for 802.11v 668
802.11v BSS transition management 668
Configure 802.11v BSS transition management (GUI) 669
Configure 802.11v BSS transition management (CLI) 669

CHAPTER 65 **802.11w** 671
Protected management frames with 802.11w 671
Integrity Group Temporal Key (IGTK) 672
Broadcast or multicast integrity protocol (BIP) 673
SA teardown protection 674
How association comeback time and SA query procedures work 674
Prerequisites for 802.11w 675
Restrictions for 802.11w 675
Configure 802.11w 675
Configure 802.11w (GUI) 675

Configure 802.11w (CLI) **676**

Disable 802.11w **677**

Monitor 802.11w **678**



Preface

This preface describes the conventions of this document and information on how to obtain other documentation. It also provides information on what's new in Cisco product documentation.

- Document Conventions , on page xxxiii
- Related Documentation, on page xxxv
- Communications, services, and additional information, on page xxxv

Document Conventions

This document uses the following conventions:

Convention	Description
^ or Ctrl	Both the ^ symbol and Ctrl represent the Control (Ctrl) key on a keyboard. For example, the key combination ^**D** or **Ctrl-D** means that you hold down the Control key while you press the D key. (Keys are indicated in capital letters but are not case sensitive.)
bold font	Commands and keywords and user-entered text appear in **bold** font.
Italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
`Courier` font	Terminal sessions and information the system displays appear in `courier` font.
`Bold Courier` font	**`Bold Courier`** font indicates text that the user must enter.
[x]	Elements in square brackets are optional.
...	An ellipsis (three consecutive nonbolded periods without spaces) after a syntax element indicates that the element can be repeated.
|	A vertical line, called a pipe, indicates a choice within a set of keywords or arguments.
[x | y]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Description
{x | y}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Reader Alert Conventions

This document may use the following conventions for reader alerts:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem.*

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time.* You can save time by performing the action described in the paragraph.

Take note of the following general safety warnings:

Warning IMPORTANT SAFETY INSTRUCTIONS

Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Read the installation instructions before using, installing, or connecting the system to the power source. Use the statement number at the beginning of each warning statement to locate its translation in the translated safety warnings for this device.

SAVE THESE INSTRUCTIONS



Related Documentation

Note Before installing or upgrading the device, refer to the release notes at https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-release-notes-list.html.

- Cisco Catalyst 9800-40 Wireless Controller documentation, located at:

  http://www.cisco.com/go/c9800

- Cisco Catalyst 9800-80 Wireless Controller documentation, located at:

  http://www.cisco.com/go/c9800

- Cisco Catalyst 9800-L Wireless Controller documentation, located at:

  http://www.cisco.com/go/c9800

Communications, services, and additional information

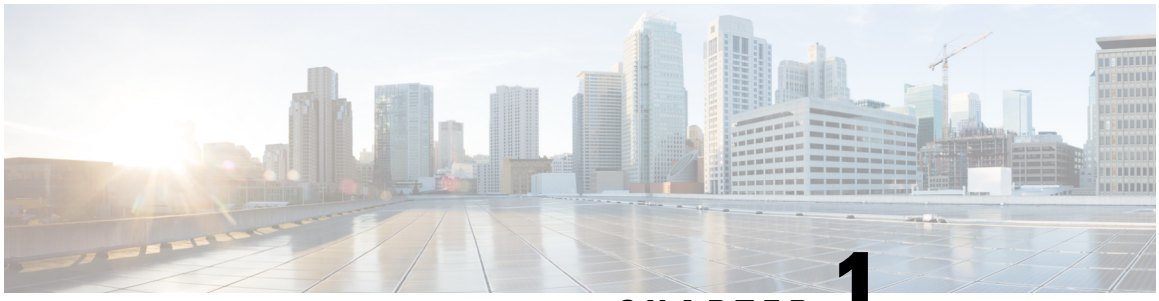
- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions, and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a gateway to the Cisco bug-tracking system, which maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. The BST provides you with detailed defect information about your products and software.

Documentation feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.



CHAPTER 1

Cisco Embedded Wireless Controller on Catalyst Access Points

Cisco Embedded Wireless Controller on Catalyst Access Points are the next generation wireless controllers that

- are IOS XE based and integrate the Radio Frequency (RF) performance excellence from Aironet systems,
- supports intent-based networking that are easy to adapt and scale, and
- are available in multiple physical and virtual form factors for flexible management options.

Available form factors and deployment options

You can deploy the controllers as physical appliances .

Management options

You can managed them the Cisco Embedded Wireless Controller on Catalyst Access Points with:

- Cisco Catalyst Center
- NETCONF or YANG, or
- GUI or CLI.

- New configuration model, tags, and profile, on page 1
- Configuration workflow, on page 2
- Initial Setup, on page 4

New configuration model, tags, and profile

The new configuration model is a wireless network configuration approach that

- uses profiles to define feature-specific parameters
- uses tags to bundle and apply profiles to APs, and
- separates policy, site, RF, and WLAN definitions for modular deployment.

Profiles define attributes such as WLAN policy, RF behavior, and AP join characteristics. Tags are logical containers that map these profiles to APs, ensuring flexible and scalable configurations.

A tag is a logical container that

- is defined by the property of policies you associate with it
- maps multiple profiles—policy, site, and RF—to an AP or client, and determines that device's configuration and behavior, and
- enables modular and flexible deployment across sites.

Every tag has a default that is created when the system boots up.

A profile is a logical container that

- include feature-specific attributes and parameters applied to tags
- represent multiple attributes related to policy, site, and RF
- determine the configuration and behavior of applied APs and associated clients, and
- is a reusable entities that can be used across tags.

Configuration workflow

Configuring wireless networks involves establishing profiles and tags that link configurations to APs for effective wireless management.

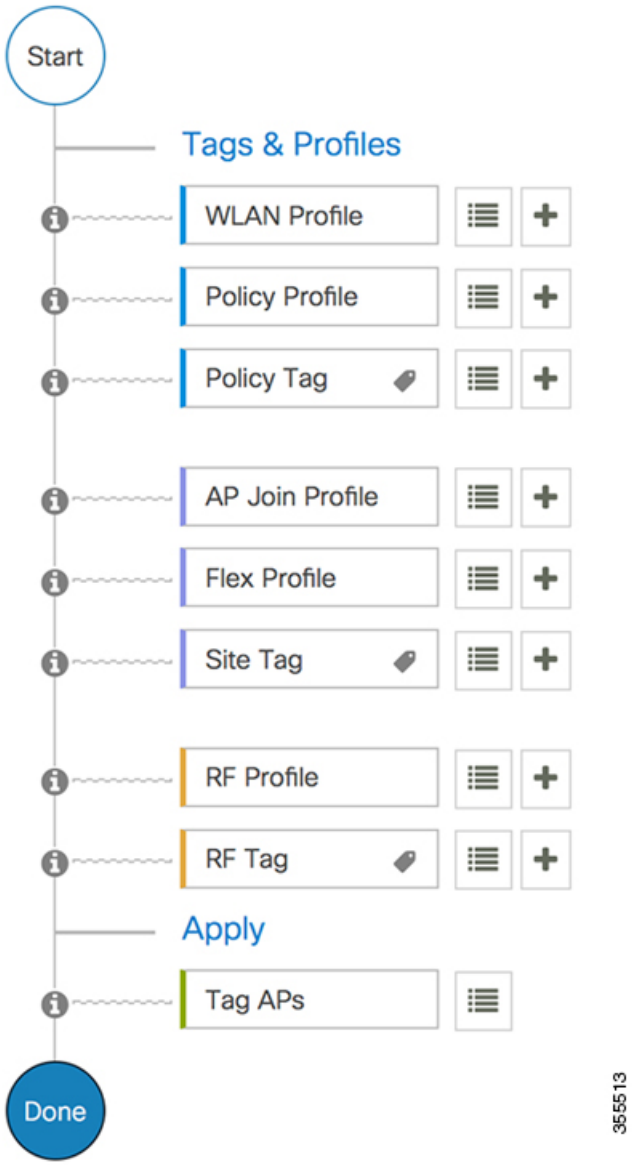
Summary

The key components involved in the process are:

- **Profiles**: Logical groupings of settings (such as WLAN, Policy, AP Join, Flex, and RF) used to configure APs.
- **Tags**: Logical containers or labels (such as Policy, Site, Tag) that help link profiles with APs.
- **APs**: Devices that implement the configurations provided by profiles and tags to deliver wireless connectivity.

Workflow

Figure 1: Configuration workflow



These are the stages of configuration:

1. **Profile creation** creates specific profiles:
    - WLAN profile for managing wireless network settings.
    - Policy profile for enforcing network rules.
    - AP Join profile for controlling AP associations.
    - Flex profile for handling local switching.
    - RF profile for optimizing radio frequency settings.

2. **Tag creation**: generates tags to complement profiles:
   - Policy tag to align with the Policy profile.
   - Site tag for specific location identification.
   - RF tag corresponding to the RF profile.
3. **Tag association**: associates the created tags with an AP to apply the configured settings and policies.

Result

The configuration workflow enables APs to be set up correctly with designated profiles and tags, resulting in efficient network operation and management.

Initial Setup

Setting Up the Controller

The initial configuration wizard in Cisco Embedded Wireless Controller on Catalyst Access Points is a simplified, out-of-the-box installation and configuration interface for the controller. This section provides instructions to set up a controller to operate in a small, medium, or large network wireless environment, where access points can join and together as a simple solution provide various services, such as corporate employee or guest wireless access on the network.



Note

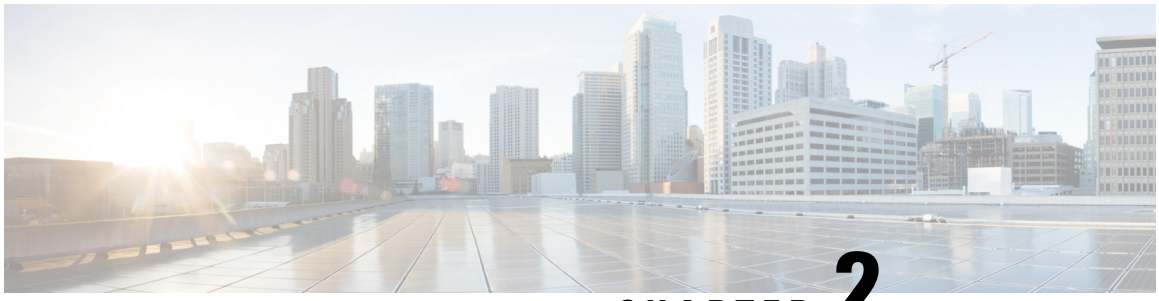
- When the AP has rebooted in the EWC mode, it broadcasts a provisioning SSID ending with the last digits of the MAC address. You can connect to the provisioning SSID using the PSK **password**.
- You can then open a browser and you are redirected to mywifi.cisco.com which takes you to the AP web UI. Enter the username as **webui** and password as **cisco**.
- We recommend that you use the **wireless ewc-ap factory-reset** command to reset the EWC device to Day0 state (with the configuration wizard). This command also resets all the APs and EWC-APs in the network to Day0 state. You can use the **erase startup-config** command to remove the configuration from the device. However, this is not synced to other devices in the network.
- After completing the Day0 wizard, the internal AP disjoins, and rejoins after one minute.
- The wireless management must be the AP Gigabit port and you cannot have several SVIs configured in IOS-XE.



PART I

System Configuration

- System Configuration, on page 7
- Smart Licensing, on page 31
- Conversion and Migration, on page 45



CHAPTER 2

System Configuration

- Information About New Configuration Model, on page 7
- Configure a wireless profile policy (GUI), on page 9
- Configure a wireless policy profile (CLI), on page 10
- Configure a flex profile, on page 11
- Configure an AP profile (GUI), on page 12
- Configure an AP profile (CLI), on page 16
- Configure an RF profile (GUI), on page 17
- Configure an RF profile (CLI), on page 17
- Configure policy tag (GUI), on page 19
- Configure a policy tag (CLI), on page 19
- Configure wireless RF tag (GUI), on page 21
- Configure wireless RF tag (CLI), on page 21
- Attach a policy tag and site tag to an AP (GUI), on page 22
- Attach policy tag and site tag to an AP (CLI), on page 23
- AP Filter, on page 24

Information About New Configuration Model

The configuration of Cisco Embedded Wireless Controller on Catalyst Access Points is simplified using different tags, namely rf-tag, policy-tag, and site-tag. The access points would derive their configuration from the profiles that are contained within the tags.

Profiles are a collection of feature-specific attributes and parameters applied to tags. The rf-tag contains the radio profiles, the policy-tag contains the WLAN profile and policy profile, and the site-tag contains the flex profile and ap-join profile.

Policy Tag

The policy tag constitutes mapping of the WLAN profile to the policy profile. The WLAN profile defines the wireless characteristics of the WLAN. The policy profile defines the network policies and the switching policies for the client (Quality of Service [QoS] is an exception which constitutes AP policies as well).

The policy tag contains the map of WLAN policy profile. There can be a maximum of 16 such entries per policy tag. Changes to the map entries are effected based on the status of the WLAN profile and policy profile. For example, if a map (WLAN1 and Policy1) is added to the policy tag, and both the WLAN profile and the policy profile are enabled, the definitions are pushed to the APs using the policy tag. However, if one of them

is in disabled state, the definition is not pushed to the AP. Similarly, if a WLAN profile is already being broadcast by an AP, it can be deleted using the no form of the command in the policy tag.

Site Tag

The site tag defines the properties of a site and contains the flex profile and the AP join profile. The attributes that are specific to the corresponding flex or remote site are part of the flex profile. Apart from the flex profile, the site tag also comprises attributes that are specific to the physical site (and hence cannot be a part of the profile that is a reusable entity). For example, the list of primary APs for efficient upgrade is a part of a site tag rather than that of a flex profile.

If a flex profile name or an AP profile name is changed in the site tag, the AP is forced to rejoin the controller by disconnecting the Datagram Transport Layer Security (DTLS) session. When a site tag is created, the AP and flex profiles are set to default values (default-ap-profile and default-flex-profile).

RF Tag

The RF tag contains the 2.4 GHz and 5 GHz RF profiles. The default RF tag contains the global configuration. Both these profiles contain the same default values for global RF profiles for the respective radios.

Profiles

Profiles are a collection of feature-specific attributes and parameters applied to tags. Profiles are reusable entities that can be used across tags. Profiles (used by tags) define the properties of the APs or its associated clients.

WLAN Profile

WLAN profiles are configured with same or different service set identifiers (SSIDs). An SSID identifies the specific wireless network for the controller to access. Creating WLANs with the same SSID allows to assign different Layer 2 security policies within the same wireless LAN.

To distinguish WLANs having the same SSID, create a unique profile name for each WLAN. WLANs with the same SSID must have unique Layer 2 security policies so that clients can select a WLAN based on the information advertised in the beacon and probe responses. The switching and network policies are not part of the WLAN definition.

Policy Profile

Policy profile broadly consists of network and switching policies. Policy profile is a reusable entity across tags. Anything that is a policy for a client that is applied on an AP or controller is moved to the policy profile, for example, VLAN, ACL, QoS, session timeout, idle timeout, AVC profile, bonjour profile, local profiling, device classification, BSSID QoS, and so on. However, all the wireless-related security attributes and features on the WLAN are grouped under the WLAN profile.

Flex Profile

Flex profile contains policy attributes and remote site-specific parameters. For example, the EAP profiles that can be used when the AP acts as an authentication server for local RADIUS server information, VLAN-ACL mapping, VLAN name-to-ID mapping, and so on.

AP Join Profile

The default AP join profile values will have the global AP parameters and the AP group parameters. The AP join profile contains attributes that are specific to AP, such as CAPWAP, IPv4 and IPv6, UDP Lite, High Availability, Retransmit config parameters, Global AP failover, Hyperlocation config parameters, Telnet and SSH, 11u parameters, and so on.



Note Telnet is not supported for the following Cisco AP models: 1542D, 1542I, 1562D, 1562E, 1562I, 1562PS, 1800S, 1800T, 1810T, 1810W,1815M, 1815STAR, 1815TSN, 1815T, 1815W, 1832I, 1840I, 1852E, 1852I, 2802E, 2802I, 2802H, 3700C, 3800, 3802E, 3802I, 3802P, 4800, IW6300, ESW6300, 9105AXI, 9105AXW, 9115AXI, 9115AXE, 9117I, APVIRTUAL, 9120AXI, 9120AXE, 9124AXI, 9124AXD, 9130AXI, 9130AXE, 9136AXI, 9162I, 9164I, and 9166I.

RF Profile

RF profile contains the common radio configuration for the APs. RF profiles are applied to all the APs that belong to an AP group, where all the APs in that group have the same profile settings.

Association of APs

APs can be associated using different ways. The default option is by using Ethernet MAC address, where the MAC is associated with policy-tag, site tag, and RF tag.

In filter-based association, APs are mapped using regular expressions. A regular expression (regex) is a pattern to match against an input string. Any number of APs matching that regex will have policy-tag, site tag, and RF tag mapped to them, which is created as part of the AP filter.

In AP-based association, tag names are configured at the PnP server and the AP stores them and sends the tag name as part of discovery process.

In location-based association, tags are mapped as per location and are pushed to any AP Ethernet MAC address mapped to that location.

Modifying AP Tags

Modifying an AP tag results in DTLS connection reset, forcing the AP to rejoin the controller. If only one tag is specified in the configuration, default tags are used for other types, for example, if only policy tag is specified, the default-site-tag and default-rf-tag will be used for site tag and RF tag.

Configure a wireless profile policy (GUI)

Define and apply a wireless profile policy to manage network behavior and access for wireless clients.

Use this task to create or modify a policy profile, ensuring wireless network policies suit your organization's needs.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click **Add**.

Step 3 In the **Add Policy Profile** window, in **General** tab, enter a name and description for the policy profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces. Do not use spaces as it causes system instability.

Step 4 To enable the policy profile, set **Status** as **Enabled**.

Step 5 In the WLAN Switching Policy section, choose the following, as required:

- No Central Switching: Tunnels both the wireless user traffic and all control traffic via CAPWAP to the centralized controller where the user traffic is mapped to a dynamic interface/VLAN on the controller. This is the normal CAPWAP mode of operation.
- Central Authentication: Tunnels client data to the controller, as the controller handles client authentication.
- No Central DHCP: The DHCP packets received from AP are centrally switched to the controller and then forwarded to the corresponding VLAN based on the AP and the SSID.
- Central Association Enable: When central association is enabled, all switching is done on the controller.
- Flex NAT/PAT: Enables Network Address Translation(NAT) and Port Address Translation (PAT) mode.

Step 6 Click **Save & Apply to Device**.

The wireless profile policy is successfully created and applied. Devices use the updated policy settings for wireless client management.

Configure a wireless policy profile (CLI)

Define and apply a wireless policy profile on your device using command-line interface commands.

Use this task to configure wireless profile policies on Cisco wireless controllers. Policy profiles specify settings such as VLAN mapping, idle timeouts, and accounting lists for wireless networks.



Note When a client moves from an old controller to a new controller (managed by Cisco Prime Infrastructure), the old IP address of the client is retained, if the IP address is learned by ARP or data gleaning. To avoid this scenario, ensure that you enable **ipv4 dhcp required** command in the policy profile. Otherwise, the IP address gets refreshed only after a period of 24 hours.

Follow the procedure given to configure a wireless profile policy:

Before you begin

- Ensure you have administrator privileges to access and configure the device.
- Have the required VLAN ID, idle timeout value, and accounting list details available, if applicable.

Follow these steps to configure a wireless profile policy:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile and enters wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy rr-xyz-policy-1
```

Step 3 (Optional) Configure the duration of idle timeout in seconds.

Example:

```
Device(config-wireless-policy)# idle-timeout 1000
```

Step 4 Configure the VLAN name or VLAN ID.

Example:

```
Device(config-wireless-policy)# vlan 24
```

Step 5 Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-wireless-policy)# no shutdown
```

Step 6 (Optional) View detailed information about a policy profile, using the **show wireless profile policy detailed** *policy-profile-name* command.

Example:

```
Device# show wireless profile policy summary
```

The wireless policy profile is configured and enabled on the device. All specified settings are applied to the selected policy.

What to do next

- Associate the policy profile with a WLAN as needed.
- Review and validate the applied settings by connecting a client device and confirming expected behavior.

Configure a flex profile

Create or modify a flex profile to customize wireless network behavior for specific sites or device groups.

Use flex profiles within your network management system to define site-specific configurations such as VLANs, SSIDs, or access policies.

Before you begin

Identify the devices or sites to apply the flex profile.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a flex profile and enter flex profile configuration mode.

Example:

```
Device(config)# wireless profile flex rr-xyz-flex-profile
```

Step 3 (Optional) Enable default parameters for the flex profile.

Example:

```
Device(config-wireless-flex-profile)# description xyz-default-flex-profile
```

Step 4 (Optional) Enable ARP caching.

Example:

```
Device(config-wireless-flex-profile)# arp-caching
```

Step 5 Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-wireless-flex-profile)# end
```

Step 6 (Optional) View detailed parameters about the flex profile, use the **show wireless profile flex detailed** *flex-profile-name* command.

Example:

```
Device# show wireless profile flex summary
```

The flex profile is created or updated and available for assignment to devices or sites.

What to do next

Assign the flex profile to the relevant site, device, or group as needed.

Configure an AP profile (GUI)

Configure and customize AP join profiles for your wireless deployment.

Use this task to define, modify, or apply AP profile parameters such as country code, LED state, timers, VLAN tagging, security settings, management options, and advanced features, using the device's graphical interface.

Before you begin

- Review the default AP join profile to update parameters for your environment (For example, Control and Provisioning of Wireless Access Points (CAPWAP), IPv4 or IPv6, UDP Lite, High Availability, retransmit configuration parameters, global AP failover, Hyperlocation configuration parameters, Telnet or SSH, 11u parameters, and so on.)
- Obtain required information, such as network-specific settings, controller addresses, credentials, and profile names.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click **Add** .

The **Add AP Join Profile** window is displayed.

Step 3 In the **General** tab, enter a name and description for the AP join profile.

Step 4 Check the **LED State** check box to set the LED state of all APs connected to the device to blink, making them easier to locate.

Step 5 In the **Client** tab and **Statistics Timer** section, enter the time in seconds that the AP sends its 802.11 statistics to the controller.

Step 6 In the **TCP MSS Configuration** section, check the **Adjust MSS Enable** check box to enter value for Adjust MSS. You can enter or update the maximum segment size (MSS) for transient packets that traverse a router. TCP MSS adjustment enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

In a CAPWAP environment, a lightweight AP discovers a device by using CAPWAP discovery mechanisms, and then sends a CAPWAP join request to the device. The device sends a CAPWAP join response to the AP that allows the AP to join the device.

When the AP joins the device, the device manages its configuration, firmware, control transactions, and data transactions.

Step 7 In the **AP** tab, you can configure various options:

- General

a) In the **General** tab, check the **Switch Flag** check box to enable switches.
b) Check the **Power Injector State** check box if power injector is being used. Use power injectors to provide flexible powering options for APs, such as local power, multiport switches with inline power, or multiport power patch panels.
c) From the **Power Injector Type** drop-down list, choose power injector type from the following options:

- Installed: If you want the AP to examine and remember the MAC address of the currently connected switch port. (This selection assumes that a power injector is connected.)
- Override: To enable the AP to operate in high-power mode without first verifying a matching MAC address.

d) In the **Injector Switch MAC** field, enter the MAC address of the switch.
e) From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST* , *EAP-TLS* , or *EAP-PEAP* .

f) From the **AP Authorization Type** drop-down list, choose the type as either *CAPWAP DTLS +* or *CAPWAP DTLS* .
g) In the **Client Statistics Reporting Interval** section, enter the interval for 5 GHz and 2.4 GHz radios in seconds.
h) Check the **Enable** check box to enable extended module.
i) From the **Profile Name** drop-down list, choose a profile name .
j) Click **Save & Apply to Device** .

- Hyperlocation: Cisco Hyperlocation is a location solution that allows to track the location of wireless clients with the accuracy of one meter. Selecting this option disables all other fields in the screen except for NTP Server.

a) In the **Hyperlocation** tab, check the **Enable Hyperlocation** check box.
b) Enter the **Detection Threshold** value to filter out packets with low RSSI. The valid range is –100 dBm to –50 dBm.
c) Enter the **Trigger Threshold** value to set the number of scan cycles before sending a BAR to clients. The valid range is 0 to 99.
d) Enter the **Reset Threshold** value to reset value in scan cycles after trigger. The valid range is 0 to 99.
e) Enter the **NTP Server** IP address.
f) Click **Save & Apply to Device** .

- BLE: If your APs are Bluetooth Low Energy (BLE) enabled, they can transmit beacon messages that are packets of data or attributes transmitted over a low-energy link. These BLE beacons are used for health monitoring, proximity detection, asset tracking, and in-store navigation. You can customize BLE beacon settings for each AP, even if they are configured globally.

a) In the **BLE** tab, enter a value in the **Beacon Interval** field to indicate how often you want your APs to send out beacon advertisements to nearby devices. The range is from one to 10, with a default value of one.
b) In the **Advertised Attenuation Level** field, enter the attenuation level. The range is from 40 to 100, with a default of 59.
c) Click **Save & Apply to Device**.

Step 8 In the **Management** tab, you can configure the following:

- Device

a) In the **Device** tab, enter the **IPv4/IPv6 Address** of the TFTP server, **TFTP Downgrade** section.
b) In the **Image File Name** field, enter the name of the software image file.
c) From the **Facility Value** drop-down list, choose the appropriate facility.
d) Enter the IPv4 or IPv6 address of the host.
e) Choose the appropriate **Log Trap Value**.
f) Enable Telnet, SSH or both configurations, if required.
g) Enable core dump, if required.
h) Click **Save & Apply to Device**.

- User

a) In the **User** tab, enter username and password details.
b) Choose the appropriate password type.
c) In the **Secret** field, enter a custom secret code.

d) Choose the appropriate secret type.
e) Choose the appropriate encryption type.
f) Click **Save & Apply to Device**.

- Credentials

a) In the **Credentials** tab, enter local username and password details.
b) Choose the appropriate local password type.
c) Enter 802.1x username and password details.
d) Choose the appropriate 802.1x password type.
e) Enter the time in seconds after which the session should expire.
f) Enable local credentials, 802.1x credentials, or both as required.
g) Click **Save & Apply to Device**.

a) In the **CDP Interface** tab, enable the CDP state, if required.
b) Click **Save & Apply to Device**.

Step 9 In the **Rogue AP** tab, check the **Rogue Detection** check box to enable rogue detection.

Step 10 In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

This field specifies the minimum RSSI value for which a Rogue AP should be reported. All Rogue APs with RSSI lower than what is configured will not be reported to controller.

Step 11 In the **Rogue Detection Transient Interval** field, enter the transient interval value.

This field indicates how long the Rogue AP should be seen before reporting the controller.

Step 12 In the **Rogue Detection Report Interval** field, enter the report interval value.

This field indicates the frequency (in seconds) of Rogue reports sent from AP to controller.

Step 13 Check the **Rogue Containment Automatic Rate Selection** check box to enable rogue containment automatic rate selection.

The AP selects the best rate for the target Rogue, based on its RSSI.

Step 14 Check the **Auto Containment on FlexConnect Standalone** check box to enable the feature.

The AP continues containment if it moves to FlexConnect standalone mode.

Step 15 Click **Save & Apply to Device**.

The AP join profile is created or updated. Devices assigned to this profile use its configuration for network operation and management.

What to do next

- Verify that APs have successfully joined and received the new settings by reviewing AP status.
- Adjust profile settings as necessary for site-specific needs or to resolve configuration issues.

Configure an AP profile (CLI)

Set up and customize an AP profile using CLI commands to apply network-wide AP configurations.

Use this procedure when you need to define or modify an AP profile for your wireless controller, typically to standardize AP settings or apply feature-specific options.

Before you begin

Identify the name and required settings for your AP profile.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile xyz-ap-profile
```

Note

- In an AP profile, the **EAP-FAST** is the default EAP type.
- When you delete a named profile, the APs associated with that profile does not revert to the default profile.

Step 3 Add a description for the AP profile.

Example:

```
Device(config-ap-profile)# description "xyz ap profile"
```

Step 4 Enable CDP for all Cisco APs.

Example:

```
Device(config-ap-profile)# cdp
```

Step 5 Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-ap-profile)# end
```

Step 6 (Optional) Display detailed information about an AP join profile.

Example:

```
Device# show ap profile name xyz-ap-profile detailed
```

The AP profile is created or updated with specified settings, and changes are active on the controller.

What to do next

Assign APs to the new profile as required, and verify AP behavior matches the intended configuration.

Configure an RF profile (GUI)

Create and enable an RF profile to optimize radio frequency settings for your wireless network.

Use this task to define an RF profile that sets parameters for radio bands and device operation. This ensures consistent performance across your wireless deployment.

Before you begin

Use the same RF profile name when configuring the wireless RF tag. If the RF tag contains an RF profile that does not exist, the radios do not operate.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **RF**.

Step 2 On the **RF Profile** window, click **Add**.

Step 3 In the **General** tab, enter a name for the RF profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 4 Choose the appropriate **Radio Band**.

Step 5 To enable the profile, set the status as **Enable**.

Step 6 Enter a **Description** for the RF profile.

Step 7 Click **Save & Apply to Device**.

The new RF profile is created, enabled, and applied to the device.

What to do next

- Verify that the RF profile settings are visible and active on your device.
- Assign the profile to RF tags or APs, if required, to complete your configuration.

Configure an RF profile (CLI)

Create and enable an RF profile to optimize radio frequency settings for your wireless network.

Use this task to define an RF profile that sets parameters for radio bands and device operation. This ensures consistent performance across your wireless deployment.

Before you begin

Use the same RF profile name when configuring the wireless RF tag. If the RF tag contains an RF profile that does not exist, the radios do not operate.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an RF profile, and enter RF profile configuration mode.

Example:

```
Device(config)# ap dot11 24ghz rf-profile rfprof24_1
```

Note
Use the **24ghz** command to configure the 802.11b parameters. Use the **5ghz** command to configure the 802.11a parameters.

Step 3 (Optional) Enable default parameters for the RF profile.

Example:

```
Device(config-rf-profile)# default
```

Step 4 Enable the RF profile on the device.

Example:

```
Device(config-rf-profile)# no shutdown
```

Step 5 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config-rf-profile)# end
```

Step 6 (Optional) Display a summary of available RF profiles.

Example:

```
Device# show ap rf-profile summary
```

Step 7 (Optional) Display detailed information about a particular RF profile.

Example:

```
Device# show ap rf-profile name rfprof24_1 detail
```

You have created and enabled a new RF profile, and applied it to the device.

What to do next

- Verify that the RF profile settings are visible and active on your device.
- Assign the profile to RF tags or APs as required to complete your configuration.

Configure policy tag (GUI)

Create and apply a policy tag to group wireless local area network (WLAN) and policy profiles for your network configuration.

Use this task when you need to define or update policy tags for your wireless network devices using the GUI.

Before you begin

- Prepare unique names for policy tags using ASCII characters (32 to 126, no leading or trailing spaces).
- Identify the WLAN and policy profiles you plan to map.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **Policy**.

Step 2 Click **Add** to view the **Add Policy Tag** window.

Step 3 Enter a name and description for the policy tag. You can use ASCII characters from 32 to 126, but do not include any leading or trailing spaces.

Step 4 Click **Add** to map WLAN and policy.

Step 5 Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.

Step 6 Click **Save & Apply to Device**.

Your device has the new policy tag applied. The mapped WLAN and policy profiles are now active based on your configuration.

What to do next

Verify that connected devices use the updated policy tag and that expected network policies are enforced.

Configure a policy tag (CLI)

Create and apply a policy tag to group wireless local area network (WLAN) and policy profiles for your network configuration.

Use this task when you need to define or update policy tags for your wireless network devices using the CLI.

Before you begin

- Prepare unique names for policy tags using ASCII characters (32 to 126, no leading or trailing spaces).
- Identify the WLAN and policy profiles you plan to map.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure policy tag and enter policy tag configuration mode.

Example:

```
Device(config-policy-tag)# wireless tag policy default-policy-tag
```

Note
When performing local web authentication, the clients connected to a controller get disconnected intermittently before session timeout.

Step 3 Add a description to a policy tag.

Example:

```
Device(config-policy-tag)# description "default-policy-tag"
```

Step 4 Map a remote-LAN profile to a policy profile.

Example:

```
Device(config-policy-tag)# remote-lan remote-lan-name policy profile-policy-name port-id
port-id-number
```

Step 5 Map a policy profile to a WLAN profile.

Example:

```
Device(config-policy-tag)# wlan wlan-name policy profile-policy-name
```

Step 6 Exit policy tag configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-policy-tag)# end
```

Step 7 (Optional) Display the configured policy tags.

Example:

```
Device# show wireless tag policy summary
```

Note
To view detailed information about a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

Your device has the new policy tag applied. The mapped WLAN and policy profiles are now active based on your configuration.

What to do next

Verify that connected devices use the updated policy tag and the expected network policies are applied.

Configure wireless RF tag (GUI)

Create and apply an RF tag that defines radio frequency profiles for device groups.

Procedure

Step 1 a) Choose **Configuration** > **Tags & Profiles** > **Tags** > **RF**.

Step 2 Click **Add** to view the **Add RF Tag** window.

Step 3 Enter a name and description for the RF tag. The name can be ASCII characters from 32 to 126 and must not include leading or trailing spaces.

Step 4 Choose the required **5 GHz Band RF Profile** and **2.4 GHz Band RF Profile** to be associated with the RF tag.

Step 5 Click **Update & Apply to Device**.

The RF tag is created with the specified RF profiles and applied to the device.

Configure wireless RF tag (CLI)

Assign and configure an radio frequency (RF) tag on your wireless device using the CLI, to manage radio profile settings per tag.

Perform this task to create or modify a wireless RF tag. This allows you to apply specific 2.4 GHz, 5 GHz, or 6 GHz RF profiles to associated APs.

Before you begin

- You can use only two profiles (2.4-GHz and 5-GHz band RF profiles) in an RF tag.
- Ensure you use the same AP tag name you created during the AP configuration task.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create an RF tag to enter wireless RF tag configuration mode.

Example:

```
Device(config)# wireless tag rf rftag1
```

Step 3 Attach an IEEE 802.11b RF policy to the RF tag.

Example:

```
Device(config-wireless-rf-tag)# 24ghz-rf-policy rfprof24_1
```

To configure a dot11a policy, use the **5ghz-rf-policy** command.

Step 4 Add a description for the RF tag.

Example:

```
Device(config-wireless-rf-tag)# description Test
```

Step 5 Exit configuration mode to return to privileged EXEC mode.

Example:

```
Device(config-wireless-rf-tag)# end
```

Step 6 Display the available RF tags.

Example:

```
Device# show wireless tag rf summary
```

Step 7 Display detailed information of a particular RF tag.

Example:

```
Device# show wireless tag rf detailed rftag1
```

You have configured the RF tag and it is now available for assignment.

What to do next

Assign the RF tag to the appropriate AP tag or APs if you have not already done so.

Attach a policy tag and site tag to an AP (GUI)

Assign a policy tag and site tag to an AP using the GUI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

The **All Access Points** section displays details of all the APs on your network.

Step 2 To edit the configuration details of an AP, select the row for that AP.

The **Edit AP** window is displayed.

Step 3 In the **General** tab and **Tags** section, specify the appropriate policy, site, and RF tags that you created on the **Configuration** > **Tags & Profiles** > **Tags** window.

Step 4 Click **Update & Apply to Device**.

The AP is now associated with the specified policy, site, and optionally RF tags you selected.

Attach policy tag and site tag to an AP (CLI)

Assign a policy tag and site tag to an AP using the CLI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a Cisco AP and enters AP profile configuration mode.

Example:

```
Device(config)# ap F866.F267.7DFB
```

Note
The *mac-address* should be a wired mac address.

Step 3 Map a policy tag to the AP.

Example:

```
Device(config-ap-tag)# policy-tag rr-xyz-policy-tag
```

Step 4 Map a site tag to the AP.

Example:

```
Device(config-ap-tag)# site-tag rr-xyz-site
```

Step 5 Associate the RF tag.

Example:

Step 6 Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-ap-tag)# end
```

Step 7 (Optional) Display AP details and the tags associated to it.

Example:

```
Device# show ap tag summary
```

Step 8 Display the AP name with tag information.

Example:

```
Device# show ap name "ap-name" tag info
```

Step 9 (Optional) Display the AP name with tag details.

Example:

```
Device# show ap name ap-name tag detail
```

The AP is now associated with the specified policy, site, and optionally, RF tags you selected

AP Filter

AP filters

An AP filter is a global tagging mechanism that

- associates APs with specific tag sources based on defined filter criteria
- organizes and applies tag sources in priority order according to configuration, and
- enables flexible control over which tags are assigned to each AP throughout the device discovery and onboarding process.

Additional reference information

AP filters function similarly to access control lists (ACLs) on the controller and operate at the global level. You can create AP filters based on AP names or other attributes, and incorporate filter criteria into discovery requests. Tag sources for APs can originate from static configuration, the AP filter engine, per-AP Plug and Play (PnP), or default configuration. The AP filter feature determines tag precedence, ensuring the correct tags are assigned according to organizational policies.

You cannot disable the AP filter feature. However, you can configure the priority of tag sources using the **ap filter-priority <priority> <filter-name>** command.

You can specify tag names at the PnP server (similarly to Flex groups or AP groups). During AP discovery and join requests, the AP stores and sends the allocated tag name as part of the request, allowing for automated and dynamic tag assignment.

Analogy: ticket sorting gate

An AP filter is like a ticket sorting gate at an event venue. Each attendee (AP) presents information at the entrance, and the sorting gate (filter) checks their details to assign them to the right group or area (tag source) based on predefined criteria and priorities. Just as event organizers use sorting gates to efficiently direct attendees, network administrators use AP filters to efficiently assign tags and manage APs in large wireless environments.

Set tag priority (GUI)

Set the order of priority for tag sources in AP configuration using the GUI

Use this task whenever you want to control which tag source is applied first to your AP configurations in the system. This helps ensure the desired tag assignment behavior.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Tag Source**.

Step 2 Drag and drop the tag sources to change priorities.

The system saves and applies the new tag source priority for AP configuration.

Set tag priority

Resolve ambiguity in tag assignment by defining the priority of tag sources for access points.

-

When an AP joins the wireless controller, it picks up tags from multiple sources. Setting tag source priorities ensures APs use tags from the preferred source. If you do not configure precedence, default priorities are used.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 (Optional) Configure AP tag source priority. Default priorities for Static, Filter, and PnP.

Example:

```
Device(config)# ap tag-source-priority 2 source pnp
```

Step 3 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 4 Revalidate AP tag sources. The priorities become active only after this command is run.

Example:

```
Device# ap tag-sources revalidate
```

Note
If you change the priorities for Filter and PnP, and need to verify the changes, run the **revalidate** command.

The controller applies the new tag source priorities, and APs assign tags based on your configured precedence.

Create an AP filter (GUI)

Define an AP filter to automatically apply tags to APs that match specific naming patterns.

Use this task to organize and manage access points by creating filter rules based on AP names and assigning tags for policy, site, or RF profiles.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter** .

Step 2 Click **Add** .

Step 3 In the **Associate Tags to AP** dialog box that is displayed, enter the **Rule Name** , the **AP name regex** , and the **Priority** .

Step 4 (Optional) You can also choose the policy tag from the **Policy Tag Name** drop-down list, the site tag from the **Site Tag Name** drop-down list and the RF tag from the **RF Tag Name** drop-down list.

Step 5 Click **Apply to Device** .

The AP filter is created. Any AP that matches the specified pattern is automatically assigned the designated tags.

Create an AP filter (CLI)

Configure an AP filter using the CLI to apply specific policy, RF, and site tags to APs matching a name pattern.

- Identify the regular expression pattern for the AP names you want to filter.
- Determine the appropriate policy, RF, and site tag names to use.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP filter.

Example:

```
Device(config)# ap filter filter-1
```

Step 3 Configure the AP filter based on regular expression.

Example:

```
Device(config-ap-filter)# ap name-regex testany
```

For example, if you have named an AP as **ap-lab-12**, then you can configure the filter with a regular expression, such as **ap-lab-\d+** , to match the AP name.

Step 4 Configure a policy tag for this filter.

Example:

```
Device(config-ap-filter)# tag policy pol-tag1
```

Step 5 Configure an RF tag for this filter.

Example:

```
Device(config-ap-filter)# tag rf rf-tag1
```

Step 6 Configure a site tag for this filter.

Example:

```
Device(config-ap-filter)# tag site site1
```

Step 7 Exit configuration mode to return to privileged EXEC mode.

Example:

```
Device(config-ap-filter)# end
```

The AP filter is configured with the specified criteria and tag associations. APs with names matching your pattern will have the corresponding tags applied.

Configure filter priority (GUI)

Configure and manage the priority of AP filters to control tag associations on APs.

Perform this task when you need to add a new AP filter or change the priority of an existing AP filter using the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter**.

Step 2 Choose:

a) To set up a new AP filter, click **Add**. In the **Associate Tags to AP** dialog box that is displayed, enter the **Rule Name**, the **AP name regex** and the **Priority**. Optionally, you can also select the **Policy Tag Name**, the **Site Tag Name** and the **RF Tag Name**. Click **Apply to Device**.

b) To update the priority of an existing AP filter, click on the filter and in the **Edit Tags** dialog box and change the **Priority**. In case the filter is inactive, you cannot configure a priority. Click **Update and Apply to Device**.

The AP filter priority is configured or updated and applied to the specified devices.

Configure filter priority (CLI)

Configure and manage the priority of AP filters to control tag associations on APs.

Perform this task when you need to add a new AP filter or change the priority of an existing AP filter using the GUI.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure AP filter priority. Valid values range from zero to 1023. The value zero represents the highest priority.

Example:

```
Device(config)# ap filter priority 10 filter-name test1
```

This step is necessary to activate the filter.

Step 3 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config-ap)# end
```

The AP filter priority is configured or updated and applied to the specified devices.

Verify AP filter configuration

These **show** commands are used to display tag sources and filters, and their priorities.

To view the tag source priorities, use the command:

```
Device# show ap tag sources

Priority Tag source
--------------------------------
0 Static
1 Filter
2 AP
3 Default
```

To view the available filters, use the command:

```
Device# show ap filter all

Filter Name             regex             Policy Tag               RF Tag              Site
 Tag
------------------------------------------------------------------------------------------
first                    abcd              pol-tag1                 rf-tag1
site-tag1
test1                   testany                                                      site1
```

```
filter1                 testany
```

To view the list of active filters, use the command:

```
Device# show ap filters active

Priority   Filter Name       regex          Policy Tag              RF Tag
Site Tag
------------------------------------------------------------------------------------------------------------
10         test1             testany
site1
```

To view the source of an AP tag, use the command:

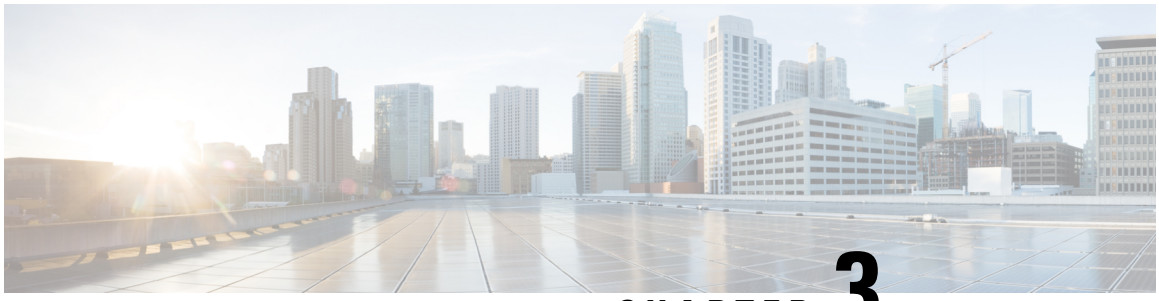
```
Device# show ap tag summary

Number of APs: 4

AP Name          AP Mac         Site Tag Name    Policy Tag Name    RF Tag Name
Misconfigured Tag Source
------------------------------------------------------------------------------------------------------------
AP002A.1034.CA78 002a.1034.ca78 named-site-tag   named-policy-tag   named-rf-tag   No Filter

AP00A2.891C.2480 00a2.891c.2480 named-site-tag   named-policy-tag   named-rf-tag   No Filter

AP58AC.78DE.9946 58ac.78de.9946 default-site-tag default-policy-tag default-rf-tag No AP
AP0081.C4F4.1F34 0081.c4f4.1f34 default-site-tag default-policy-tag default-rf-tag No Default
```



CHAPTER 3

Smart Licensing

- Information About Cisco Smart Licensing, on page 31
- Creating a Smart Account, on page 33
- Using Smart Licensing, on page 34
- Using Specified License Reservation (SLR), on page 34
- Enabling Specified License Reservation in CSSM, on page 35
- Enabling Smart Software Licensing, on page 36
- Registering for Smart License (Connected Mode), on page 37
- Enabling Smart License Reservation, on page 38
- Enabling Smart Call Home Reporting, on page 38
- Configuring AIR License Level (GUI), on page 39
- Configuring AIR License Level (CLI), on page 40
- Configuring AIR Network Essentials License Level, on page 40
- Configuring AIR Network Advantage License Level , on page 41
- Verifying Smart Licensing Configurations, on page 41

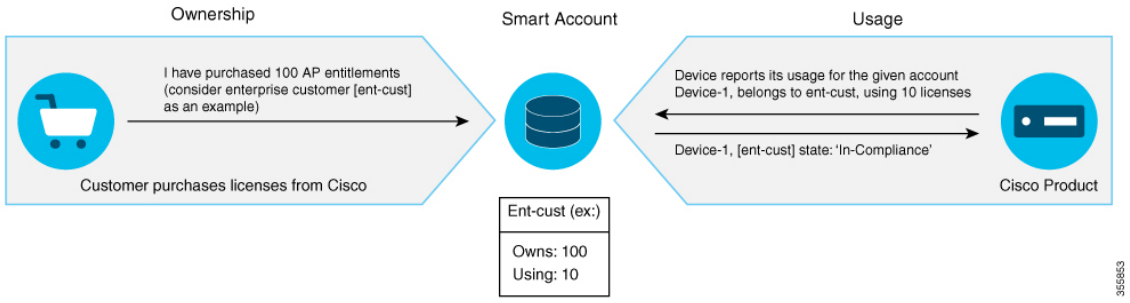
Information About Cisco Smart Licensing

Cisco Smart Licensing is a flexible licensing model that provides you with an easier, faster, and more consistent way to purchase and manage software across the Cisco portfolio and across your organization. And it's secure – you control what users can access. With Smart Licensing you get:

- Easy Activation: Smart Licensing establishes a pool of software licenses that can be used across the entire organization—no more PAKs (Product Activation Keys).
- Unified Management: Cisco License Central provides a complete view into all of your Cisco products and services in an easy-to-use portal, so you always know what you have and what you are using.
- License Flexibility: Your software is not node-locked to your hardware, so you can easily use and transfer licenses as needed.

To use Smart Licensing, you must first set up a Smart Account on Cisco License Central. For a more detailed overview on Cisco Licensing, visit the Licensing Hub.

Figure 2: Relationship Between Ownership, Smart Account, and Usage



Note Starting with Cisco IOS XE Gibraltar 16.12.1, the Cisco Catalyst 9800 Series Wireless Controller does not support satellite server for licensing reporting. You should use the Cisco Smart Software Manager (CSSM) for any licensing reporting.

Once your product is registered in CSSM, you will be able to view the license usage using your Smart Account or Virtual Account for every eight hours.

Note

- Smart Licensing registration is lost when the device switches from controller to autonomous mode and back. In such instances, you should re-register the controller to CSSM to restore licenses authorization.
- After adding new license in the Cisco Smart Software Manager (CSSM) for customer virtual account, run the **license smart renew auth** command on the controller to get the license status changed from Out OF Compliance to Authorised.

Access points support the following AIR licensing levels:

- AIR Network Essential (AIR-NE)
- AIR Network Advantage (AIR-NA)
- AIR DNA Essential (AIR-DNA-E)
- AIR DNA Advantage (AIR-DNA-A)

Note The *AIR-DNA-A* and *AIR-DNA-E* are the available license levels on the controller.

The AIR-DNA-A is the default mode.

You can configure as *AIR-DNA-A* or *AIR-DNA-E* license level and on term expiry, you can move to the Network Advantage or Network Essentials license level, if you do not want to renew the DNA license.

Smart Licensing Reservation Types

License reservation is a mechanism to reserve node locked licenses and install them on the controller.

The following are the license reservation types:

- Permanent License Reservation (PLR)—All licenses are reserved.
- Specified License Reservation (SLR)—Only specific licenses are reserved. Supports term licenses.

The controller supports four different entitlement registration or reporting on Smart Licensing or service reservation. Every connecting AP requires a Cisco DNA Center License to leverage the unique value properties of the controller.

Note The controller boots up with *AIR-DNA-A* as the default. Any change in the license level requires a reboot.

Entitlement Reporting

Entitlement reporting is nothing but reporting the number of access points on the controller to the Cisco Smart Software Manager (CSSM).

The entitlement reporting is based on the configured AIR license level on the controller.

Note Two types of entitlement reporting occurs when you are in *AIR-DNA-E* and *AIR-DNA-A* levels. For instance, if your controller reports 100 APs as count, your entitlement reporting displays *100 AIR-NE* and *100 AIR-DNA-E*. Similarly, it also displays *100 AIR-NA* and *100 AIR-DNA-A* to CSSM.

Creating a Smart Account

Procedure

Step 1 Navigate to the Cisco Software Central web page:

https://software.cisco.com/#

The Cisco Software Central page is displayed.

Step 2 From the **Important News** pop-up window, click **Get a Smart Account**.

(Or)

From the **Administration** area, click **Request a Smart Account**.

Follow the process to create a Smart Account.

Note
You need to have a Smart Account to use Smart Licensing.

You set up a Smart Account on Cisco License Central. For a more detailed overview on Cisco Licensing, visit the Licensing Hub.

Using Smart Licensing

Before you begin

Follow the procedure given below to cover the high-level steps on how to use smart licensing:

Procedure

Step 1 Configure your device for smart licensing.

Step 2 Login to **CSSM** customer **Smart Account** > **Virtual Account** to generate a token.

Step 3 Execute the following command on your device:

```
Device# license smart register idtoken <token-id>
```

Note
You can get the *token-id* from the CSSM web portal.

Note
You can use the **license smart register idtoken** *token-id* **force** command to register the device again even if the same device was registered with CSSM earlier.

Using Specified License Reservation (SLR)

Procedure

Step 1 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 **license smart reservation**

Example:

```
Device(config)# license smart reservation
```

Enables specified license reservation mode on the controller.

Step 3 **license smart reservation request local**

Example:

```
Device(config)# license smart reservation request local
```

Generates a request code.

Note
Enter this request code in the Cisco Smart Software Manager portal:

```
CB-ZL-AIR-9500C-K9:9J4FVHMBXCO-BjSeUVwmn-D8
```

Step 4 **end**

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Enabling Specified License Reservation in CSSM

Before you begin

You should have a smart account and virtual account to generate the authorization code for the controller.

Procedure

Step 1 Login to CSSM.

Step 2 De-register the smart license, if the controller is reporting to a satellite server.

```
Device(config)# license smart deregister
```

Step 3 Enable Specified License Reservation in the controller.

```
Device(config)# license smart reservation
```

Step 4 Verify the license reservation status on your controller using the following command:

```
Device# show license reservation

License reservation: ENABLED

Overall status:
  Active: PID:C9800-CL-K9,SN:9PQFKND9ZR8
      Reservation status: NOT INSTALLED
      Export-Controlled Functionality: NOT ALLOWED
  Standby: PID:C9800-CL-K9,SN:9UD8BBTHL1S
      Reservation status: NOT INSTALLED
      Export-Controlled Functionality: NOT ALLOWED
```

Step 5 Generate *request code* on your controller using the following command:

```
Device(config)# license smart reservation request all
```

```
Request code for active : CG-ZC9800-CL-K9:9PQFKND9ZR8-BjSeUVwmn-8E
Request code for standby : CG-ZC9800-CL-K9:9PQFKND9ZR8-BjSeUVwmn-8E
```

Option *all* will generate the request code for both active and stand-by, if the controller is in HA pair.

Option *local* will generate the request code for active or standalone controller.

Step 6 Generate *authorization code*, using the *request code*, for each controller separately in CSSM and install both the codes in the controller. You can install the *authorization code* of standby controller thorough active controller.

a) Go to CSSM and navigate to your Smart Account and Virtual Account: https://software.cisco.com/software/csws/ws/platform/home#SmartLicensing-Inventory
b) Click **Licenses** tab.
c) Click **License Reservation** button and enter the request code obtained from the previous step in to the **Reservation Request Code** field.
d) Click **Next**.
e) In the **Select Licenses** tab, select the **Reserve a specific license** radio button and enter the number of licenses required to reserve in the **Reserve** text box.
f) Click **Next**.
g) In the **Review and Confirm** tab, check the quantity and license type, and click **Generate Authorization Code** button.
h) From the **Authorization Code** tab, select **Download as File** option to download the **authorization code**.

Note
Repeat **Step b** to **Step h** to generate *authorization code* for the standby controller.

Step 7 Upload the *authorization code* file to the controller bootflash: directory.

```
Device# copy ftp://<ip-address>authorization-code.txt bootflash:
Destination filename [authorization-code.txt]
```

Step 8 Install the *authorization code* file in the controller using the following command.

```
Device# license smart reservation install file authorization-code.txt
```

Note
Use the same command to install the *authorization code*for stand-by controller also using active controller in case of HA.

Step 9 Verify the license summary after installing the *authorization code* on your controller using the following command:

```
Device# show license summary
```

Enabling Smart Software Licensing

Procedure

Step 1 Navigate to the Cisco Software Central web page using the following link:

https://software.cisco.com/#

The Cisco Software Central page is displayed.

Step 2 From the **License** tab, click **Smart Software Licensing**.

The Smart Software Licensing page is displayed.

Step 3 Click the **Inventory** tab to view **Virtual Account: Accounting** page details.

Step 4 Click **New Token** to register the product instances to this virtual account.

The Create Registration Token page is displayed.

Step 5 In the **Description** field, enter a description for the ID token.

Step 6 Check the **Allow export-controlled functionality on the products registered with this token** checkbox to enable export-controlled functionality.

Step 7 Click **Create Token**.

Note
Licenses cannot be purchased with the wireless controller. All licenses can be purchased with access points.

Registering for Smart License (Connected Mode)

Procedure

	Command or Action	Purpose
Step 1	Copy the token generated in *Enabling Smart Software Licensing*.	
Step 2	On Catalyst ME, use this token to register to Smart License.	
Step 3	**license smart register idtoken** *token* **Example:** `Device(config)# license smart register idtoken CB-ZL-AIR-9500C-K9:9J4FVHMBXCO-BjSeUVwmn-D8`	Registers for Smart License using the token number.
Step 4	**license smart register idtoken** *token* **[force]** **Example:** `Device(config)# license smart register CB-ZL-AIR-9500C-K9:9J4FVHMBXCO-BjSeUVwmn-D8 force`	Enables registering for Smart License with force if normal registering does not take place.
Step 5	**license smart deregister** **Example:**	De-registers the smart license.

	Command or Action	Purpose
	`Device(config)# license smart deregister`	

Enabling Smart License Reservation

Procedure

	Command or Action	Purpose
Step 1	**license smart reservation**	Enables Smart License Reservation on ME. Request code is generated.
Step 2	Navigate to the Cisco Software Central web page using the following link:	https://software.cisco.com/# The Cisco Software Central page is displayed.
Step 3	From the **License** tab, click **Smart Software Licensing**.	The Smart Software Licensing page is displayed.
Step 4	Click the **Inventory** tab and then click **Licenses** > **License Reservation**.	Follow steps 1-4 to **Enter Request Code**, **Select Licenses**, **Review and Confirm**, and to generate an **Authorization Code**.
Step 5	After generating the Authorization Code, Click **Download as File** option to save to AuthorizationCode.txt.	The Authorization Code is saved as a text file.
Step 6	On Catalyst ME, copy the authorization file to flash (/bootflash/AutorizationCode.txt).	
Step 7	**license smart reservation install file** *filename* **Example:** `Device(config)# license smart reservation install file AuthorizationCode.txt`	Installs the reserved licenses using the Authorization Code file..
Step 8	**license smart reservation return**	Returns the reserved licenses.
Step 9	**no license smart reservation**	Exits from the SLR mode.

Enabling Smart Call Home Reporting

Procedure

Step 1 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 **call-home reporting contact-email-addr** *email-address* **http-proxy** *proxy-server port-number*

Example:

```
Device(config)# call-home reporting contact-email-addr sample@cisco.com http-proxy 120.20.2.2
 5
```

Enables Call Home reporting.

- *port-number*—The valid range is from 1 to 65535.

Step 3 **end**

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

For more information on Smart Call Home, see:

https://www.cisco.com/c/en/us/td/docs/switches/lan/smart_call_home/book/SCH31_Ch3.html

Configuring AIR License Level (GUI)

Procedure

Step 1 Choose **Administration** > **Licensing**.

Step 2 Click **Change Wireless License Level**. The **Change Wireless License Level** dialog box is displayed.

Step 3 Select the License Level using the drop-downs.

Step 4 After changing the **New Level** values, click **Save & Reload** (Or) **Save without Reload**. Alternatively, you can click **Reload** to reload the device. During this time, you will lose network connectivity to the device. If you wish to continue, click **Yes**.

Step 5 Click refresh icon to refresh the device.

Configuring AIR License Level (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 2	**license air level** {**air-network-advantage** | **air-network-essentials**}
	Example:
	`Device(config)# license air level air-network-advantage`
	`Device(config)# license air level air-network-essentials`
		Configures AIR license level.
		• air-network-advantage—Is the AIR network advantage license level.
		• air-network-essentials—Is the AIR network essential license level.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring AIR Network Essentials License Level

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**license air level network-essentials addon air-dna-essentials** **Example:** `Device(config)# license air level network-essentials addon air-dna-essentials`	Configures AIR network essentials license level.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring AIR Network Advantage License Level

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**license air level air-network-advantage addon air-dna-advantage** **Example:** `Device(config)# license air level air-network-advantage addon air-dna-advantage`	Configures AIR network advantage license level.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Smart Licensing Configurations

To verify the smart licensing status and license usage, use the following command:

```
Device# show license all
Smart Licensing Status

======================
Smart Licensing is ENABLED
Registration:
  Status: REGISTERED
  Smart Account: BU Production Test
  Virtual Account: MEWLC-DE
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Nov 19 15:36:51 2018 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: May 18 15:36:50 2019 UTC
  Registration Expires: Nov 19 15:31:24 2019 UTC

License Authorization:
  Status: AUTHORIZED on Nov 20 07:46:48 2018 UTC
  Last Communication Attempt: SUCCEEDED on Nov 20 07:46:48 2018 UTC
  Next Communication Attempt: Dec 20 07:46:48 2018 UTC
  Communication Deadline: Feb 18 07:40:49 2019 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Utility:
  Status: DISABLED
```

```
Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

License Usage
==============
AP Perpetual Networkstack Essentials (DNA_NWSTACK_E):
  Description: AP Perpetual Network Stack entitled with DNA-E
  Count: 2
  Version: 1.0
  Status: AUTHORIZED
  Export status: NOT RESTRICTED


Product Information
===================
UDI: PID:AIR-AP1900I-B-K9,SN:9ICMGRNU4U0

Agent Version
=============
Smart Agent for Licensing: 4.6.0_rel/2
Component Versions: SA:(1_3_dev)1.0.15, SI:(dev22)1.2.1, CH:(rel5)1.0.3, PK:(dev18)1.0.3

Reservation Info
================
License reservation: DISABLED
```

To verify the smart licensing status, use the following command:

```
Device# show license status
Smart Licensing is ENABLED

Utility:
Status: DISABLED
Data Privacy:
Sending Hostname: yes
Callhome hostname privacy: DISABLED
Smart Licensing hostname privacy: DISABLED
       Version privacy: DISABLED

Transport:
Type: Callhome

Registration:
Status: REGISTERED
Smart Account: BU Production Test
Virtual Account: MEWLC-DE
   Export-Controlled Functionality: Allowed

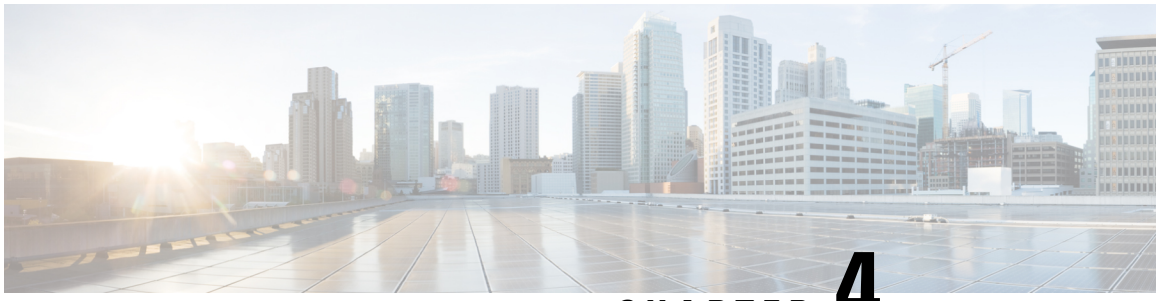
Initial Registration: SUCCEEDED on Nov 19 15:36:51 2018 UTC
Last Renewal Attempt: None
Next Renewal Attempt: May 18 15:36:51 2019 UTC
Registration Expires: Nov 19 15:31:25 2019 UTC

License Authorization:
Status: AUTHORIZED on Nov 19 16:23:42 2018 UTC
Last Communication Attempt: SUCCEEDED on Nov 19 16:23:42 2018 UTC
Next Communication Attempt: Dec 19 16:23:42 2018 UTC
Communication Deadline: Feb 17 16:18:17 2019 UTC
```

```
Export Authorization Key:
Features Authorized:
<none>
```

To verify the air license level and smart licensing status, use the following command:

```
Device# show version
AIR License Level: AIR DNA Advantage
Next reload AIR license Level: AIR DNA Advantage
Smart Licensing Status: REGISTERED/AUTHORIZED
 cisco AIR-AP1900I-B-K9 (VXE) processor (revision VXE) with 322620K bytes of memory.
```



CHAPTER 4

Conversion and Migration

- Conversion and Migration in Embedded Wireless Controller Capable APs , on page 45
- Types of Conversion, on page 45
- Access Point Conversion, on page 46
- Network Conversion, on page 49
- SKU Conversion Scenarios, on page 51
- Converting AireOS Mobility Express Network to Embedded Wireless Controller Network , on page 52

Conversion and Migration in Embedded Wireless Controller Capable APs

The Cisco Embedded Wireless Controller on Catalyst Access Points is not supported on any non-802.11ax (non-11ax) based access points (AP). It is only supported on 802.11ax (11ax) based APs. The embedded wireless controller is the only supported form of Cisco Mobility Express on 11ax based APs.

The conversion enables you to convert the 11ax APs running CAPWAP to embedded wireless controller and vice-versa.

Types of Conversion

The types of conversion scenarios supported are:

- AP Conversion – The following AP conversions are supported:
  - Converting a CAPWAP AP to Embedded Wireless Controller - This conversion is required when you have an AP with a CAPWAP image, and you want to use the AP to deploy a embedded wireless controller based network. In order to do this, you must convert the CAPWAP AP to a embedded wireless controller.
  - Converting an Embedded Wireless Controller AP to a CAPWAP AP – This conversion is required if you want to migrate the APs from an embedded wireless controller network to a non-embedded wireless controller network; or if you do not want the APs to participate in the primary AP election process.
- Network Conversion

- SKU Conversion

Access Point Conversion

This section gives the details of converting a CAPWAP access point to anembedded wireless controller.

Converting a CAPWAP AP to an Embedded Wireless Controller Capable AP

Note Before converting from CAPWAP to embedded wireless controller (EWC), ensure that you upgrade the corresponding AP with the CAPWAP image in Cisco AireOS Release 8.10.105.0. If this upgrade is not performed, the conversion will fail.

To convert an 802.11ax AP with a CAPWAP image to an embedded wireless controller capable image, either download the controller image based on the automated image download process, use the conversion command, or convert through the WebUI.

Note When the AP is embedded wireless controller capable, the AP can participate in the primary AP election process. Only if the AP is elected as a primary, can it perform the controller functionality.

Converting an Embedded Wireless Controller Capable AP to a CAPWAP AP

To convert an 802.11ax AP from the embedded wireless controller network to a non-embedded wireless controller network, set the AP type to CAPWAP using the conversion command or the WebUI, respectively, and then plug it on to the controller network so that it joins the controller. If the image on the controller is different from the image on the AP, a new CAPWAP image is requested from the controller.

Converting a Single AP to CAPWAP or Embedded Wireless Controller Capable AP (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `>enable`	Enters privileged EXEC mode.
Step 2	**wireless ewc ap ap-type** *ap-name* {**capwap** | **ewc**} **Example:**	Changes the AP to CAPWAP type or to the embedded controller type.

	Command or Action	Purpose
	`Device#wireless ewc-ap ap ap-type ap-name capwap`	

Example

```
wireless ewc-ap ap ap-type ap-name {capwap | ewc}
```

AP Conversion Deployment Scenarios

1. Standalone 802.11ax CAPWAP AP to start an embedded wireless controller network:

802.11ax AP	Embedded Wireless Controller Capable APs	Use-Case	Automatic Conversion
Standalone 802.11ax CAPWAP AP	Network does not exist.	To use a the standalone 802.11ax CAPWAP AP as the first AP for setting up the embedded wireless controller network.	Automatic conversion is not possible. You must download both the controller and the AP image using the supported image transfer protocols with AP command: `ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>]`

2. Non-802.11ax CAPWAP AP joining an existing embedded wireless controller network:

CAPWAP AP	Embedded Wireless Controller Capable APs	Use-Case	Automatic Conversion
CAPWAP AP - Neither AireOS-Mobility Express capable, or, embedded wireless controller capable AP, or, AireOS-Mobility Express capable Wave 2 APs.	Existing network	To bring in a CAPWAP AP which is not embedded wireless controller capable, into an existing embedded wireless controller network, to add one more AP to the existing network.	Yes, automatic conversion is possible. This is automatically taken care through the AP Join image download process.

3. 802.11ax AP joining an existing embedded wireless controller network:

Embedded Wireless Controller Capable AP	Embedded Wireless Controller Network	Use-Case	Automatic Conversion

802.11ax AireOS-CAPWAP AP or 802.11ax Catalyst CAPWAP AP or 802.11ax embedded wireless controller capable AP	Existing network	To bring in an 802.11ax AP from an AireOS-CAPWAP network, or, a CAPWAP network, or, from another embedded wireless controller network into an existing embedded wireless controller network, to add one more AP to the existing network.	Yes, automatic conversion takes place.
			This is automatically taken care through the AP Join image download process.
			If the AP type is explicitly set to CAPWAP, then the AP continues to act as a CAPWAP AP unless it is converted back again to embedded wireless controller AP using the AP command, Controller command, or the WebUI.
			The following command is used for conversion as well as AP image download:
			`ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip>Controller ImagePath>]`
			The following command is used to convert a specific AP to CAPWAP or embedded wireless controller:
			`wireless ewc-ap ap ap-type` *`ap-name`* `{capwap | ewc-ap}`

4. 802.11ax embedded wireless controller AP joining an AireOS CAPWAP network or a CAPWAP network:

802.11 AX Embedded Wireless Controller Capable AP	Embedded Wireless Controller Network	Use-Case	Automatic Conversion

802.11ax AP which was earlier an embedded wireless controller AP	Existing network	To bring an existing 802.11ax embedded wireless controller AP and add it to the CAPWAP network or the AireOS-CAPWAP network to add one more AP to the existing network.	It is recommended to convert the AP to CAPWAP type before bringing it to the CAPWAP network. This conversion can be done manually by using the AP command, the Controller command, Controller WebUI, or by using the DHCP option.
			After conversion, the normal image download process should be followed.
			`ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip>Controller ImagePath>]`
			`wireless ewc-ap ap ap-type ap-name {capwap | ewc-ap}`

Network Conversion

This section describes network conversion thorugh the conversion command and the network conversion deployment scenarios.

Converting the Network (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `>enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**Wireless ewc-ap ap capwap** *primary-controller-name* {**A:B:C:D** | **X:X:X:X::X**} **Example:** `Device#wireless ewc-ap ap capwap wlc-name 10.0.0.0`	Specifies the wireless controller name and IP address to which all the APs currently connected to the embedded wireless controller network should join.

Network Conversion Deployment Scenarios

1. Converting an existing centralized CAPWAP network or AireOS CAPWAP network to the embedded wireless controller network

Existing Network	Embedded Wireless Controller Network	Use-Case	Automatic Conversion
CAPWAP Network: Centralized CAPWAP network or AireOS-CAPWAP network with at least one 802.11ax AP.	Network does not exist.	To convert the existing centralized CAPWAP network or the AireOS-CAPWAP network to theembedded wireless controller network.	No, automatic conversion does not take place. You need to pick one 802.11ax AP to download both the controller and AP image using the supported image transfer protocols with the AP command. `ap-type {capwap | ewc-ap} <sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>]`

2. Converting an existing embedded wireless controller network to an AireOS CAPWAP network or to a centralized CAPWAP network

Existing Network	Embedded Wireless Controller Network	Use-Case	Automatic Conversion
Embedded wireless controller network with many APs.	Existing network	To convert the existing embedded wireless controller network to an AireOS-CAPWAP network or to a centralized CAPWAP network.	No automatic conversion. You must convert all the APs or one AP at a time using the controller command to specify the IP address of the controller to which the AP has to join. You can also use the WebUI to convert the selected APs or all the APs by specifying the IP address of the controller to which the AP has to join.

SKU Conversion Scenarios

1. 802.11ax Embedded Wireless Controller SKU instead of CAPWAP SKU

SKU	Network	Use-Case	Automatic Conversion
802.11ax embedded wireless controller SKU instead of CAPWAP SKU	Network does not exist.	For an order placed for 802.11ax embedded wireless controller SKU instead of CAPWAP SKU, it should be converted to CAPWAP SKU.	No automatic conversion available. You can use **DHCP option 43** to point to the Catalyst 9800 controller so that the APs join the Catalyst 9800 controller as a CAPWAP AP.

	SKU	Network	Use-Case	Automatic Conversion
2.	802.11ax CAPWAP SKU instead of the embedded wireless controller SKU.	Network does not exist.	For an order placed for the 802.11ax CAPWAP SKU instead of the embedded wireless controller SKU and now would like to convert it to embedded wireless controller SKU.	No automatic conversion available. You should pick one 802.11ax AP to download both the controller and AP image using the supported image transfer protocols with AP command.`ap-type ewc-ap <sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>`

Converting AireOS Mobility Express Network to Embedded Wireless Controller Network

Procedure

Step 1 Remove the **Next Preferred Master** configuration from the existing AireOS Mobility Express network and save the configuration.

Step 2 Power down all the APs in the AireOS Mobility Express network including the primary AP.

Step 3 Power-on the 11 AX AP with the embedded wireless controller SKU so that it launches the controller.

Step 4 Provision the 11 AX AP with the required configuration (if the box is in Day-0, provision the mandatory configuration to get to Day-1).

Step 5 Copy, Translate, and Apply all the AireOS Mobility Express configurations to the 11 AX embedded wireless controller AP, add image download configuration.

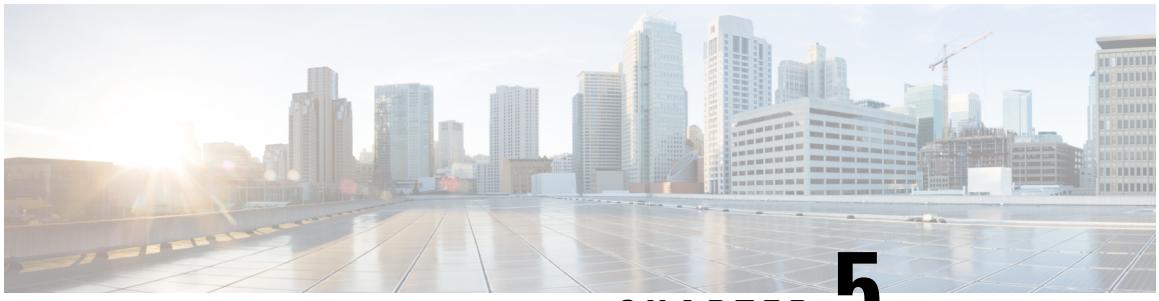
Step 6 Power-on all the APs in the AireOS Mobility Express network. All the APs from the earlier AireOS Mobility Express network will join as regular APs in the embedded wireless controller network.



PART II

Lightweight Access Points

- Country Codes, on page 55
- AP Priority, on page 61
- Rogue per AP, on page 63
- 802.11 Parameters for Cisco Access Points, on page 71
- 802.1x Support, on page 85



CHAPTER 5

Country Codes

- Information About Country Codes, on page 55
- Prerequisites for Configuring Country Codes, on page 56
- Configure country codes (GUI), on page 56
- How to Configure Country Codes , on page 56
- Configuration examples for configuring country codes, on page 58

Information About Country Codes

Controllers and access points are designed for use in many countries with varying regulatory requirements. The radios within the access points are assigned to a specific regulatory domain at the factory (such as -E for Europe), but the country code enables you to specify a particular country of operation within that regulatory domain (such as FR for France or ES for Spain). Configuring a country code ensures that each radio's broadcast frequency bands, interfaces, channels, and transmit power levels are compliant with country-specific regulations.

Information About Japanese Country Codes

Country codes define the channels that can be used legally in each country. These country codes are available for Japan:

- JP: Allows only –J radios to join the controller
- J2: Allows only –P radios to join the controller
- J3: Uses the –U frequencies, but allows –U, –P, and –Q (other than 1550/1600/2600/3600) radios to join the controller
- J4: Allows 2.4G JPQU and 5G PQU to join the controller.

Note The 1550, 1600, 2600, and 3600 APs require J4.

See the Channels and Maximum Power Settings for Cisco Aironet Lightweight Access Points document for the list of channels and power levels supported by access points in the Japanese regulatory domains.

Prerequisites for Configuring Country Codes

- Generally, you should configure one country code per device; you configure one code that matches the physical location of the device and its access points. You can configure up to 20 country codes per device. This multiple-country support enables you to manage access points in various countries from a single device.
- When the multiple-country feature is used, all the devices that are going to join the same RF group must be configured with the same set of countries, configured in the same order.
- Access points are capable of using all the available legal frequencies. However, access points are assigned to the frequencies that are supported in their relevant domains.
- The country list configured on the RF group leader determines which channels the members will operate on. This list is independent of which countries have been configured on the RF group members.
- For devices in the Japan regulatory domain, you must have at least one access point with a -J regulatory domain joined to your device.
- You cannot delete any country code using the configuration command **wireless country country-code** if the specified country was configured using the **ap country list** command and vice-versa.

Configure country codes (GUI)

Set the allowed country codes on wireless APs to comply with local regulations.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points** > **Country**.

Step 2 On the **Country** page, check the check box for each country where your APs are installed. If you select more than one check box, a message displays that RRM channels and power levels are limited to common channels and power levels.

Step 3 Click **Apply**.

How to Configure Country Codes

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**show wireless country supported** **Example:** `Device# show wireless country supported`	Displays a list of all the available country codes.
Step 3	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 4	**ap dot11 24ghz shutdown** **Example:** `Device(config)# ap dot11 24ghz shutdown`	Disables the 802.11b/g network.
Step 5	**ap dot11 5ghz shutdown** **Example:** `Device(config)# ap dot11 5ghz shutdown`	Disables the 802.11a network.
Step 6	**ap country** *country_code* **Example:** `Device(config)# ap country IN`	
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**show wireless country channels** **Example:** `Device# show wireless country channels`	Displays the list of available channels for the country codes configured on your device. **Note** Perform Steps 9 through 17 only if you have configured multiple country codes in Step 6.
Step 9	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 10	**no ap dot11 5ghz shutdown** **Example:** `Device(config)# no ap dot11 5ghz shutdown`	Enables the 802.11a network.
Step 11	**no ap dot11 24ghz shutdown** **Example:** `Device(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b/g network.

	Command or Action	Purpose
Step 12	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 13	**ap name** *cisco-ap* **shutdown** **Example:** `Device# ap name AP02 shutdown`	Disables the access point. **Note** Ensure that you disable only the access point for which you are configuring country codes.
Step 14	**ap name** *cisco-ap* **no shutdown** **Example:** `Device# ap name AP02 no shutdown`	Enables the access point.

Configuration examples for configuring country codes

View channel list for country codes

These examples show how to display the list of available channels for the country codes on your devicedevice:

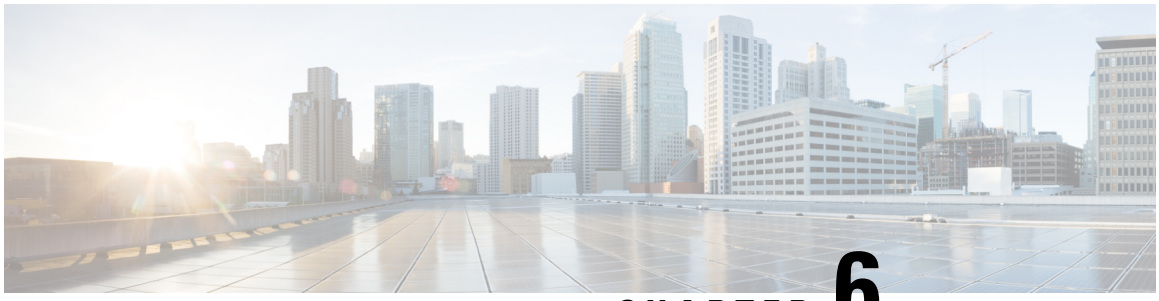
```
DeviceDevice# show wireless country channels

Configured Country.......................: US - United States
KEY: * = Channel is legal in this country and may be configured manually.
     A = Channel is the Auto-RF default in this country.
     . = Channel is not legal in this country.
     C = Channel has been configured for use by Auto-RF.
     x = Channel is available to be configured for use by Auto-RF.
  (-,-) = (indoor, outdoor) regulatory domain allowed by this country.
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11bg         :
Channels         :                  1 1 1 1 1
                  1 2 3 4 5 6 7 8 9 0 1 2 3 4
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US    : A * * * * A * * * * A . . .
Auto-RF          : . . . . . . . . . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11a          :                        1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Channels         :3 3 3 4 4 4 4 4 5 5 6 6 0 0 0 1 1 2 2 2 3 3 4 4 5 5 6 6
                  4 6 8 0 2 4 6 8 2 6 0 4 0 4 8 2 6 0 4 8 2 6 0 9 3 7 1 5
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US    : . A . A . A . A A A A A * * * * * . . . * * * A A A A*
Auto-RF          : . . . . . . . . . . . . . . . . . . . . . . . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
4.9GHz 802.11a   :
Channels         :                  1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2
                  1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
US (-A ,-AB )    : * * * * * * * * * * * * * * * * * * * A * * * * * A
Auto-RF          : . . . . . . . . . . . . . . . . . . . . . . . . . .

Device# show wireless country configured
```

```
Configured Country..................... US - United States
Configured Country Codes
US - United States 802.11a Indoor,Outdoor/ 802.11b Indoor,Outdoor/ 802.11g Indoor,Outdoor
```

View channel list for country codes



CHAPTER 6

AP Priority

- Failover Priority for Access Points, on page 61
- Set AP priority (GUI), on page 61
- Set AP priority (CLI), on page 62

Failover Priority for Access Points

Each embedded controller has a defined number of communication ports for access points. When multiple embedded controllers with unused access point ports are deployed on the same network and one controller fails, the dropped access points automatically poll for unused controller ports and associate with them.

The following are some guidelines for configuring failover priority for access points:

- You can configure your wireless network so that the backup embedded controller recognizes a join request from a higher-priority access point, and if necessary, disassociates a lower-priority access point as a means to provide an available port.
- Failover priority is not in effect during the regular operation of your wireless network. It takes effect only if there are more association requests after an embedded controller failure than there are available backup controller slots.
- You can enable failover priority on your network and assign priorities to the individual access points.
- By default, all access points are set to priority level 1, which is the lowest priority level. Therefore, you need to assign a priority level only to those access points that warrant a higher priority.

This section contains the following subsections:

Set AP priority (GUI)

Assign a failover priority to an AP to ensure high availability in the event of a controller failure.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 Click the Access Point.

Step 3 In the **Edit AP** dialog box, go to the **High Availability** tab.

Step 4 Select the priority from the **AP failover priority** drop-down list.

Step 5 Click **Update and Apply to Device**.

Set AP priority (CLI)

Assign a priority level to an AP to determine the order in which APs connect and operate using commands.

Note Priority of APs ranges from one to four, with four being the highest.

Procedure

Step 1 Specify the priority of an AP.

Example:

```
Device# ap name ap-name AP44d3.ca52.48b5 priority 1
```

Step 2 Display common information for all APs.

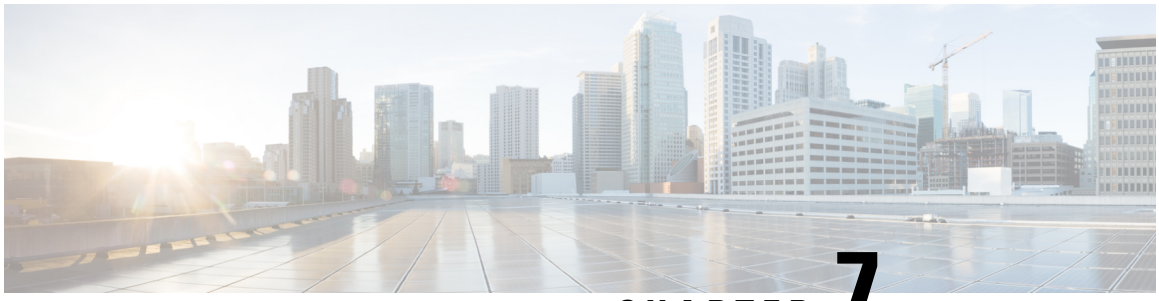
Example:

```
Device# show ap config general
```

Step 3 Display the configuration of a particular AP.

Example:

```
Device# show ap name ap-name AP44d3.ca52.48b5 config general
```



CHAPTER 7

Rogue per AP

- Rogue per AP, on page 63
- Enabling Rogue Detection, on page 64

Rogue per AP

Rogue detection is configured per AP or for a group of APs. The rogue AP detection is configured under the AP profile. The rogue AP detection configuration enabled by default and is part of the default AP profile.

The following commands are deprecated from this release:

- **wireless wps rogue detection enable**
- **wireless wps rogue detection report-interval** *interval*
- **wireless wps rogue detection min-rssi** *rssi*
- **wireless wps rogue detection min-transient-time** *transtime*
- **wireless wps rogue detection containment flex-connect**
- **wireless wps rogue detection containment auto-rate**

Enabling Rogue Detection

The following are the high-level steps to enable rogue detection:

- Configure an AP Profile
- Define a Wireless Site Tag and Assign the AP Profile
- Associate the Wireless Site Tag to an AP

Note The controller may not report the original min-rssi value due to conversions made by the AP and the controller. Hence, the reported min-rssi may be different from the original value.

Enabling Rogue Detection

Configure an AP profile (GUI)

Configure and customize AP join profiles for your wireless deployment.

Use this task to define, modify, or apply AP profile parameters such as country code, LED state, timers, VLAN tagging, security settings, management options, and advanced features, using the device's graphical interface.

Before you begin

- Review the default AP join profile to update parameters for your environment (For example, Control and Provisioning of Wireless Access Points (CAPWAP), IPv4 or IPv6, UDP Lite, High Availability, retransmit configuration parameters, global AP failover, Hyperlocation configuration parameters, Telnet or SSH, 11u parameters, and so on.)
- Obtain required information, such as network-specific settings, controller addresses, credentials, and profile names.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click **Add**.

The **Add AP Join Profile** window is displayed.

Step 3 In the **General** tab, enter a name and description for the AP join profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 4 Check the **LED State** check box to set the LED state of all APs connected to the device to blink, making them easier to locate.

Step 5 In the **Client** tab and **Statistics Timer** section, enter the time in seconds that the AP sends its 802.11 statistics to the controller.

Step 6 In the **TCP MSS Configuration** section, check the **Adjust MSS Enable** check box to enter value for Adjust MSS. You can enter or update the maximum segment size (MSS) for transient packets that traverse a router. TCP MSS adjustment enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

In a CAPWAP environment, a lightweight AP discovers a device by using CAPWAP discovery mechanisms, and then sends a CAPWAP join request to the device. The device sends a CAPWAP join response to the AP that allows the AP to join the device.

When the AP joins the device, the device manages its configuration, firmware, control transactions, and data transactions.

Step 7 In the **AP** tab, you can configure these options:

- General

a) In the **General** tab, check the **Switch Flag** check box to enable switches.

b) Check the **Power Injector State** check box if power injector is being used. Use power injectors to provide flexible powering options for APs, such as local power, multiport switches with inline power, or multiport power patch panels.

Power Injector Selection parameter enables you to protect your switch port from an accidental overload if the power injector is inadvertently bypassed.

c) From the **Power Injector Type** drop-down list, choose power injector type from these options:

- Installed—This option examines and remembers the MAC address of the currently connected switch port and assumes that a power injector is connected. Choose this option if your network contains older Cisco 6-Watt switches and you want to avoid possible overloads by forcing a double-check of any relocated APs.

  If you want to configure the switch MAC address, enter the MAC address in theInjector Switch MAC Address text box. If you want the AP to find the switch MAC address, leave theInjector Switch MAC Address text box blank.

  Note
  Each time an AP is relocated, the MAC address of the new switch port fails to match the remembered MAC address, and the AP remains in low-power mode. You must then physically verify the existence of a power injector and reselect this option to cause the new MAC address to be remembered.

- Override—This option allows the AP to operate in high-power mode without first verifying a matching MAC address. You can use this option if your network does not contain any older Cisco 6-W switches that could be overloaded if connected directly to a 12-W AP. The advantage of this option is that if you relocate the AP, it continues to operate in high-power mode without any further configuration. The disadvantage of this option is that if the AP is connected directly to a 6-W switch, an overload occurs.

d) In the**Injector Switch MAC** field, enter the MAC address of the switch .
e) From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST* , *EAP-TLS* , or *EAP-PEAP*.
f) From the **AP Authorization Type** drop-down list, choose the type as either *CAPWAP DTLS +* or *CAPWAP DTLS*.
g) In the **Client Statistics ReportingInterval** section, enter the interval for 5 GHz and 2.4 GHz radios in seconds.
h) Check the **Enable** check box to enable extended module.
i) From the **Profile Name** drop-down list, choose a profile name .
j) Click **Save & Apply to Device**.

- Hyperlocation: Cisco Hyperlocation is a location solution that allows to track the location of wireless clients with the accuracy of one meter. Selecting this option disables all other fields in the screen, except NTP Server.

a) In the **Hyperlocation** tab, check the **Enable Hyperlocation** check box.
b) Enter the **Detection Threshold** value to filter out packets with low RSSI. The valid range is –100 dBm to –50 dBm.
c) Enter the **Trigger Threshold** value to set the number of scan cycles before sending a BAR to clients. The valid range is zero to 99.
d) Enter the **Reset Threshold** value to reset value in scan cycles after trigger. The valid range is zero to 99.
e) Enter the **NTP Server** IP address.
f) Click **Save & Apply to Device**.

- BLE: If your APs are Bluetooth Low Energy (BLE) enabled, they can transmit beacon messages that are packets of data or attributes transmitted over a low energy link. These BLE beacons are frequently used for health monitoring, proximity detection, asset tracking, and in-store navigation. For each AP, you can customize BLE Beacon settings configured globally for all APs.

a) In the **BLE** tab, enter a value in the **BeaconInterval** field to indicate how often you want your APs to send out beacon advertisements to nearby devices. The range is from one to 10, with a default value of one.
b) In the **Advertised Attenuation Level** field, enter the attenuation level. The range is from 40 to 100, with a default of 59.
c) Click **Save & Apply to Device**.

- Packet Capture: Packet Capture feature allows to capture the packets on the AP for the wireless client troubleshooting. The packet capture operation is performed on the AP by the radio drivers on the current channel on which it is operational, based on the specified packet capture filter.

a) In the **Packet Capture** tab, choose an **AP Packet Capture Profile** from the drop-down list.
b) You can also create a new profile by clicking the **+** sign.
c) Enter a name and description for the AP packet capture profile.
d) Enter the **Buffer Size**.
e) Enter the **Duration**.
f) Enter the **Truncate Length** information.
g) In the **Server IP** field, enter the IP address of the TFTP server.
h) In the **File Path** field, enter the directory path.
i) Enter the username and password details.
j) From the **Password Type** drop-down list, choose the type.
k) In the **Packet Classifiers** section, use the option to select or enter the packets to be captured.
l) Click **Save**.
m) Click **Save & Apply to Device**.

Step 8 In the **Management** tab, you can configure these options:

- Device

a) In the **Device** tab, enter the **IPv4/IPv6 Address** of the TFTP server, **TFTP Downgrade** section.
b) In the **Image File Name** field, enter the name of the software image file.
c) From the **Facility Value** drop-down list, choose the appropriate facility.
d) Enter the IPv4 or IPv6 address of the host.
e) Choose the appropriate **Log Trap Value**.
f) Enable Telnet, SSH or both configurations, if required.
g) Enable core dump, if required.
h) Click **Save & Apply to Device**.

- User

a) In the **User** tab, enter username and password details.
b) Choose the appropriate password type.
c) In the **Secret** field, enter a custom secret code.
d) Choose the appropriate secret type.
e) Choose the appropriate encryption type.

f) Click **Save & Apply to Device**.

- Credentials

a) In the **Credentials** tab, enter local username and password details.
b) Choose the appropriate local password type.
c) Enter 802.1x username and password details.
d) Choose the appropriate 802.1x password type.
e) Enter the time in seconds after which the session should expire.
f) Enable local credentials, 802.1x credentials, or both as required.
g) Click **Save & Apply to Device**.

a) In the **CDPInterface** tab, enable the CDP state, if required.
b) Click **Save & Apply to Device**.

Step 9 In the **Rogue AP** tab, check the **Rogue Detection** check box to enable rogue detection.

Step 10 In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

This field specifies the minimum RSSI value for which a Rogue AP should be reported. All Rogue APs with RSSI lower than what is configured will not be reported to controller.

Step 11 In the **Rogue Detection TransientInterval** field, enter the transient interval value.

This field indicates how long the Rogue AP should be seen before reporting the controller.

Step 12 In the **Rogue Detection ReportInterval** field, enter the report interval value.

This field indicates the frequency (in seconds) of Rogue reports sent from AP to controller.

Step 13 Check the **Rogue Containment Automatic Rate Selection** check box to enable rogue containment automatic rate selection.

The AP selects the best rate for the target Rogue, based on its RSSI.

Step 14 Check the **Auto Containment on FlexConnect Standalone** check box to enable the feature.

The AP continues containment if it moves to FlexConnect standalone mode.

Step 15 Click **Save & Apply to Device**.

The AP join profile is created or updated. Devices assigned to this profile use its configuration for network operation and management.

What to do next

- Verify that APs have successfully joined and received the new settings by reviewing AP status.
- Adjust profile settings as necessary for site-specific needs or to resolve configuration issues.

Configure an AP Profile

Follow the procedure given below to configure an AP profile:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters the ap profile configuration mode.
Step 3	**description** *ap-profile-name* **Example:** `Device(config-ap-profile)# description "xyz ap profile"`	Adds a description for the ap profile.
Step 4	**rogue detection enable** **Example:** `Device(config-ap-profile)# rogue detection enable`	Enables rogue detection for individual access points. Rogue detection is enabled by default. Use this command if rogue detection is disabled.
Step 5	**rogue detection report-interval** *interval* **Example:** `Device(config-ap-profile)# rogue detection report-interval 12`	Specifies the time interval, in seconds, at which APs should send the rogue detection report to the embedded controller. The default value for *interval* is 10.
Step 6	**rogue detection min-rssi** *rssi* **Example:** `Device(config-ap-profile)# rogue detection min-rssi -128`	Specifies the minimum RSSI value that rogues should have for APs to detect them. The minimum RSSI value is –128.
Step 7	**rogue detection min-transient-time** *transtime* **Example:** `Device(config-ap-profile)# rogue detection min-transient-time 120`	Specifies the time interval at which rogues have to be consistently scanned for by APs after the first time the rogues are scanned. The lowest value for minimum transient time is 0.
Step 8	**rogue detection containment flex-connect** **Example:** `Device(config-ap-profile)# rogue detection containment flex-connect`	Sets the auto containment options for standalone FlexConnect access points. By default, this option is disabled.
Step 9	**rogue detection containment auto-rate** **Example:** `Device(config-ap-profile)# rogue detection containment auto-rate`	Sets the auto rate for containment of rogues. By default, auto-rate is disabled.

Define a Wireless Site Tag and Assign an AP Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the **Tags** page, click the **Site** tab and click **Add**.

Step 3 In the **Add Site Tag** window, enter the name in the **name** field.

Step 4 Choose the AP profile from the **AP Join Profile** drop-down list.

Step 5 Click **Save & Apply to Device**.

Define a Wireless Site Tag and Assign an AP Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**wireless tag site***site-tag* **Example:** `Device(config)# wireless tag site default-site-tag`	Enters the wireless site tag configuration mode.
Step 3	**ap-profile** *ap-profile* **Example:** `Device(config-site-tag)# ap-profile xyz-ap-profile`	Assigns an AP profile to the wireless site.
Step 4	**exit** **Example:** `Device(config-site-tag)# exit`	Returns to the global configuration mode.

Associating Wireless Tag to an AP (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 Click **AP** tab to configure the following:

- Tag Source
- Static
- Filter

Step 3 In the **Static** tab, click **Add** to perform the following:

a) Enter a MAC address.
b) Choose the appropriate **Policy Tag Name**, **Site Tag Name**, and **RF Tag Name**.
c) Click **Save & Apply to Device**.

Step 4 In the **Filter** tab, click **Add** to perform the following:

a) Enter a rule and AP name.
b) Use the slider to enable **Active**.
c) Enter the priority. The valid range is from 0 to 127.
d) Choose the appropriate **Policy Tag Name**, **Site Tag Name**, and **RF Tag Name**.
e) Click **Save & Apply to Device**.

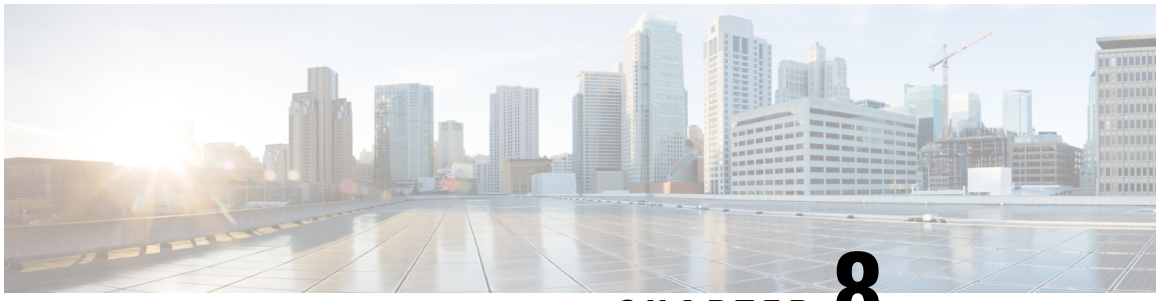
Associate Wireless Tag to an AP (CLI)

Follw the procedure given below to apply the rogue configuration defined under ap profile to the AP.

Note If the AP is not explicitly associated to a non-default site tag, it will be associated to default-site-tag and resultantly the default-ap-profile rogue configuration will be used.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**ap** *mac-address* **Example:** `Device(config)# ap F866.F267.7DFB`	Configures Cisco APs and enters the ap configuration mode.
Step 3	**site-tag** *site-tag-name* **Example:** `Device(config-ap-tag)# site-tag sitetag1`	Maps a wireless site tag to the AP.



CHAPTER 8

802.11 Parameters for Cisco Access Points

- 2.4-GHz radio support, on page 71
- 5-GHz radio support, on page 73
- Dual-band radios in Cisco AP models, on page 75
- Receiver only Dual-Band radio support, on page 81
- Configure client steering (CLI), on page 83
- Verify Cisco APs with dual-band radios, on page 84

2.4-GHz radio support

Configure 2.4-GHz radio support for the specified slot number (CLI)

Enable and configure 2.4-GHz radio features for a specified AP slot using commands.

Note The term **802.11b radio** or **2.4-GHz radio** are used interchangeably.

Procedure

Step 1 Enter the privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enable Spectrum Intelligence (SI) for the dedicated 2.4-GHz radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 SI
```

For more information, see the *Spectrum Intelligence* section in this guide. In this context, ***0*** refers to the Slot ID.

Step 3 Configure 802.11b antenna hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 antenna {ext-ant-gain antenna_gain_value |
selection [ internal | external ]}
```

- **ext-ant-gain** : configures the 802.11b external antenna gain. *antenna_gain_value* : the external antenna gain value, specified in multiples of 0.5 dBi units. The valid range is from zero to 40, with a maximum gain of 20 dBi.
- **selection** : configures the 802.11b antenna to use either the internal or external option.

Note

- For APs that support self-identifying antennas (SIA), the gain depends on the antenna and not the AP model. The AP automatically learns the gain, so no controller configuration is necessary.
- For APs that do not support SIA, the AP sends the antenna gain in the configuration payload. The default antenna gain depends on the AP model.
- Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, they do not automatically detect SIA antennas or apply the correct external gain.

Step 4 Configure beamforming for the 2.4-GHz radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 beamforming
```

Step 5 Configure advanced 802.11 channel assignment parameters for the 2.4-GHz radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 channel {channel_number | auto}
```

Step 6 Enable CleanAir for 802.11b radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 cleanair
```

Step 7 Configure 802.11n antenna for 2.4-GHz radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 dot11n antenna {A | B | C | D}
```

Here, are the descriptions of various options:

A : A is the antenna port A.

B : B is the antenna port B.

C : C is the antenna port C.

D : D is the antenna port D.

Step 8 Disable 802.11b radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 shutdown
```

Step 9 Configure transmit power level for 802.11b radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 txpower {tx_power_level | auto}
```

- *tx_power_level* : is the transmit power level in dBm. The valid range is from one to eight.
- **auto** : enables auto-RF.

5-GHz radio support

Configure 5-GHz radio support for the specified slot number (CLI)

Enable and configure 5-GHz radio features on a selected AP slot.

Note The term **802.11a radio** or **5-GHz radio** are used interchangeably in this document.

Procedure

Step 1 Enable Spectrum Intelligence (SI) for the dedicated 5-GHz radio hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 SI
```

Here, ***1*** refers to the Slot ID.

Step 2 Configure external antenna gain for 802.11a radios for a specific AP hosted on slot 1.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna ext-ant-gain antenna_gain_value
```

antenna_gain_value : refers to the external antenna gain value in multiples of .5 dBi units. The valid range is from zero to 40, with the maximum gain of 20 dBi.

Note

- For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, not on the AP model. The AP learns the gain, so controller configuration is not required.
- For APs that do not support SIA, the AP sends the antenna gain in the configuration payload. The default antenna gain depends on the AP model.
- Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, these APs do not automatically detect SIA antennas or add the correct external gain.

Step 3 Configure the antenna mode for 802.11a radios for a specific AP hosted on slot 1.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna mode {omni | sectorA | sectorB}
```

Step 4 Configure the antenna selection for 802.11a radios for a specific AP hosted on slot 1.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna selection {internal | external}
```

Step 5 Configure beamforming for the 5-GHz radio hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 beamforming
```

Step 6 Configure advanced 802.11 channel assignment parameters for the 5-GHz radio hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 channel {channel_number | auto | width [20 |
 40 | 80 | 160]}
```

Here,

channel_number : refers to the channel number. The valid range is from one to 173.

Step 7 Enable CleanAir for 802.11a radio hosted on slot 1 for a given or specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 cleanair
```

Step 8 Configure 802.11n for 5-GHz radio hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 dot11n antenna {A | B | C | D}
```

Here, are the descriptions of various options:

A : A is the antenna port A.

B : B is the antenna port B.

C : C is the antenna port C.

D : D is the antenna port D.

Change the channel hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 rrm channel 2
```

Here, *channel*: refers to the new channel created using 802.11h channel announcement. The valid range is from one to 173, provided 173 is a valid channel in the country where the AP is deployed.

Step 9 Disable 802.11a radio hosted on slot 1 for a specific AP and configure 802.11a radio hosted on slot 1 for a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 shutdown
```

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 txpower {tx_power_level | auto}
```

- *tx_power_level* : is the transmit power level in dBm. The valid range is from one to eight.

- **auto** : enables auto-RF.

Dual-band radios in Cisco AP models

A dual-band radio is a device category that

- offers connectivity on more than one frequency band (such as 2.4 GHz and 5 GHz)
- provides flexibility in network configuration, and
- is used in multiple Cisco AP models like the 2800, 3800, 4800, and 9120 series.

Feature history table

Table 1: Feature history table for dual-band radios in Cisco AP models

Feature name	Release information	Feature description
Dual-band (XOR) radio support	Cisco IOS XE 16.10.1	The Dual-band (XOR) radio support offers the ability to serve 2.4 GHz or 5 GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Key features of dual-band radios in Cisco APs

The XOR radio support can be steered manually or automatically.

- Manual steering of a band on a radio: The band on the XOR radio can only be changed manually.
- Automatic client and band steering on radios: This is managed by the Flexible Radio Assignment (FRA) feature that monitors and changes the band configurations as per site requirements.

Client steering

When a radio moves between bands (from 2.4 GHz to 5 GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5 GHz band, client steering algorithms contained in the FRA algorithms are used to steer a client between the same band co-resident radios

Cisco APs and dual-band radios

Cisco 2800, 3800, 4800, and 9120 series AP models are equipped with dual-band (XOR) radios. These models have

- radios that operate on either 2.4 GHz or 5 GHz bands, or
- the ability to passively monitor both the bands on the same AP.

These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Cisco AP models up to the Cisco 9120 APs are designed to support dual 5 GHz band operations with the *i* model supporting a dedicated Macro or Micro architecture and the *e* and *p* models supporting Macro or Macro. The Cisco 9130AXI APs support dual 5 GHz operations as Macro or Micro cell.

Limitations

These are the limitations for XOR radio support:

- RF measurement is disabled when a static channel is configured on slot 1. As a result, the dual-band radio slot 0 operates only with 5 GHz radios and not in the monitor mode. When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4 GHz radio.
- Only one of the 5 GHz radios can operate in the UNII band (100 to 144), due to an AP limitation to maintain the power budget within the regulatory limit.

Configure default XOR radio support (CLI)

Complete this task to configure the basic settings for XOR radio support.

Before you begin

Note The default radio points to the XOR radio hosted on slot 0.

Procedure

Step 1 Enable the priviledged EXEC mode.

Example:

```
Device> enable
```

Step 2 Configure the 802.11 dual-band external antenna gain on the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain antenna_gain_value
```

Enter external antenna gain value in multiple of .5 dBi units (that is, an integer value 4 means 4 x 0.5 = 2 dBi of gain).

Step 3 Shut down the default dual-band radio.

Example:

```
Device# ap name ap-name  dot11 dual-band shutdown
```

Use the **no** form of the command to enable the radio.

Step 4 Switch to the client-serving mode of the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band role manual client-serving
```

Step 5 Switch to the 2.4 GHz radio band.

Example:

```
Device# ap name ap-name  dot11 dual-band band 24ghz
```

The basic configuration of the XOR radio is completed.

Example

```
Device> enable
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain 2
Device# ap name ap-name  dot11 dual-band shutdown
Device# Device# ap name ap-name  dot11 dual-band role manual client-serving
Device# Device# ap name ap-name  dot11 dual-band band 24ghz
```

What to do next

Continue with the configuration steps listed in the "Configure advanced settings" to complete the configuration.

Configure advanced settings (CLI)

Complete this task to configure the advanced settings for XOR radio support, such as, transmit power, channel, channel width, CleanAir settings and so on.

Procedure

Step 1 Configure the transmit power for the radio on the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band txpower {transmit_power_level | auto}
```

Note
When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio.

If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.

Step 2 Configure the channel number for the dual band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel 40
```

The range of the *channel-number* is from 1 to 173.

Step 3 (Optional) Enable the auto channel assignment for the dual-band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel auto
```

Step 4 Configure the channel width for the dual band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel width {20 MHz | 40 MHz | 80 MHz | 160 MHz}
```

Step 5 Enable the Cisco CleanAir management on the dual-band radio.

Example:

```
Device# ap name ap-name  dot11 dual-band cleanair
```

Step 6 Configure a band for the Cisco CleanAir feature.

Example:

```
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
```

Selects a band for the Cisco CleanAir feature.

Use the **no** form of this command to disable the Cisco CleanAir feature.

Step 7 Configure the 802.11n dual-band antenna and antenna ports selection for a specific AP.

Example:

```
Device# ap name ap-name  dot11 dual-band dot11n antenna {A | B | C | D}
```

Here, A, B, C, and D are the antenna ports.

The configuration of the additional features for dual-band (XOR) radios are completed.

Example

```
Device# ap name ap-name  dot11 dual-band txpower auto
Device# ap name ap-name  dot11 dual-band channel 40
Device# ap name ap-name  dot11 dual-band channel width 40 MHz
Device# ap name ap-name  dot11 dual-band cleanair
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
Device# ap name ap-name  dot11 dual-band dot11n antenna A
```

Configure XOR radio support for the specified slot number (GUI)

Complete this task to configure XOR radio for the specified slot number.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, it displays the antenna PID and antenna design specifics.

Step 3 Click **Configure**.

Step 4 In the **General** tab, set the **Admin Status** as required.

Step 5 Set the **CleanAir Admin Status** field to Enable or Disable.

Step 6 Click **Update & Apply to Device**.

The XOR radio support for the specified slot number has been configured.

Configure XOR radio support for the specified slot number (CLI)

Complete this task to configure XOR radio for a specified slot number.

Procedure

Step 1 Enable the priviledged EXEC mode.

Example:

```
Device> enable
```

Step 2 Configure dual-band antenna for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 antenna ext-ant-gain
external_antenna_gain_value
```

Here, *external_antenna_gain_value* is the external antenna gain value in multiples of .5 dBi unit. The range is from 0 to 40.

Note

- For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration.
- For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.

Step 3 Configure the current band for XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 band {24ghz | 5ghz}
```

Step 4 Configures dual-band channel for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 channel {channel-number | auto | width [160
 | 20 | 40 | 80]}
```

The range of the *channel_number* is from 1 to 165.

Step 5 Enable the Cisco CleanAir feature for dual-band radios hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 cleanair band {24Ghz | 5ghz}
```

Step 6 Configure 802.11n dual-band parameters hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 dot11n antenna {A | B | C | D}
```

Here are the antenna configurations.

A : Enables antenna port A.

B : Enables antenna port B.

C : Enables antenna port C.

D : Enables antenna port D.

Step 7 Configure dual-band role for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 role {auto | manual [client-serving | monitor]}
```

Here are the dual-band roles.

- **auto** : refers to the automatic radio role selection.
- **manual** : refers to the manual radio role selection.

Note
When Flexible Radio Assignment (FRA) is disabled, XOR radios whose roles were changed by Radio Resource Management (RRM) might not automatically revert to their default radio role. To restore an individual AP to the default client-serving role, use the these commands:

```
Device# ap name ap-name dot11 dual-band slot 0 shutdown

Device# ap name ap-name dot11 dual-band slot 0 role manual client-serving

Device# ap name ap-name dot11 dual-band slot 0 band 24ghz

Device# ap name ap-name no dot11 dual-band slot 0 shutdown
```

Step 8 Disable dual-band radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name [no] dot11 dual-band slot 0 shutdown
```

Use the **no** form of this command to enable the dual-band radio.

Step 9 Configure dual-band transmit power for XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 txpower {tx-power-level | auto}
```

Here,

- *tx_power_level* : is the transmit power level in dBm. The range is from 1 to 8.
- **auto** : enables auto-RF.

The XOR radio for the specified slot number is configured.

Example

```
Device> enable
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna ext-ant-gain 2
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair band 24Ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n antenna A
                Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2
```

Verify XOR radio support

To verify the XOR radio support parameters, use these **show** commands:

Verify auto-RF information

To display the auto-RF information for a Cisco AP, use the **show ap name** *ap-name* **auto-rf dot11 dual-band** command.

```
Device# show ap name cisco-ap-name auto-rf dot11 dual-band
```

Verify the list of BSSIDs

To display the list of BSSIDs for a Cisco AP, use the **show ap name** *ap-name* **wlan dot11 dual-band** command.

Receiver only Dual-Band radio support

Receiver only dual-band radio support

A receiver only dual-band radio is a WLAN radio type that

- configures the dual-band Rx-only radio features for an AP with dual-band radios
- serves dedicated functions such as analytics, hyperlocation, wireless security monitoring, and BLE Angle of Arrival (AoA), and
- continually remains in monitor mode, preventing any *tx-rx* configuration for client or data traffic.

Configure receiver only Dual-Band parameters for APs

Enable CleanAir with receiver Only dual-band radio on a Cisco AP (GUI)

Enable CleanAir functionality on a Cisco AP that is configured with a receiver only dual-band radio through the GUI.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.

Step 3 In the **General** tab, enable the **CleanAir** toggle button.

Step 4 Click **Update & Apply to Device**.

Enabling CleanAir with receiver only dual-band radio on a Cisco AP (CLI)

Enable CleanAir spectrum analysis for a dual-band receiver-only radio on a specified Cisco AP.

Procedure

Step 1 Enter the privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enable CleanAir with receiver only (Rx-only) dual-band radio on a specific AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 cleanair band 24Ghz
Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 cleanair band 24Ghz
```

Here, 2 refers to the slot ID. Use the ***no*** form of this command to disable CleanAir.

Disable receiver only dual-band radio on a Cisco AP (GUI)

Prevent the AP from operating a receiver-only dual-band radio by using the GUI.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.

Step 3 In the **General** tab, disable the **CleanAir Status** toggle button.

Step 4 Click **Update & Apply to Device**.

Disable receiver only dual-band radio on a Cisco AP (CLI)

Disable the receiver only dual-band radio function on a specific Cisco AP using commands.

Procedure

Step 1 Enter the privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Disable receiver only dual-band radio on a specific Cisco AP.

Example:

```
Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 shutdown
Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 shutdown
```

Here, 2 refers to the slot ID. Use the ***no*** form of this command to enable receiver only dual-band radio.

Configure client steering (CLI)

Enable and customize client steering on a wireless device to optimize client distribution between macro (5 GHz) and micro (2.4 GHz) radios, improving wireless network performance and load balancing.

Before you begin

Enable Cisco CleanAir on the corresponding dual-band radio.

Procedure

Step 1 Configure the macro-micro client load–balancing window for a set number of clients.

Example:

```
Device(config)# wireless macro-micro steering transition-threshold balancing-window
number-of-clients(0-65535) 10
```

Step 2 Configure the macro-micro client parameters for a minimum client count for transition.

Example:

```
Device(config)# wireless macro-micro steering transition-threshold client count
number-of-clients(0-65535) 10
```

Step 3 Configure the macro-to-micro transition RSSI.

Example:

```
Device(config)# wireless macro-micro steering transition-threshold macro-to-micro RSSI-in-dBm(
 -128—0) -100
```

Step 4 Configure the micro–to–macro transition RSSI.

Example:

```
Device(config)# wireless macro-micro steering transition-threshold micro-to-macro
RSSI-in-dBm(-128—0) -110
```

Step 5 Configure the number of probe cycles to be suppressed.

Example:

```
Device(config)# wireless macro-micro steering probe-suppression aggressiveness
number-of-cycles(-128—0) -110
```

Step 6 Configure the macro-to-micro probe in RSSI.

Example:

```
Device(config)# wireless macro-micro steering probe-suppression hysteresis RSSI-in-dBm -5
```

The range is between –6 to –3.

Step 7 Enable probe suppression mode.

Example:

```
Device(config)# wireless macro-micro steering probe-suppression probe-only
```

Step 8 Enable probe and single authentication suppression mode.

Example:

```
Device(config)# wireless macro-micro steering probe-suppression probe-auth
```

Step 9 Display the wireless client steering information.

Example:

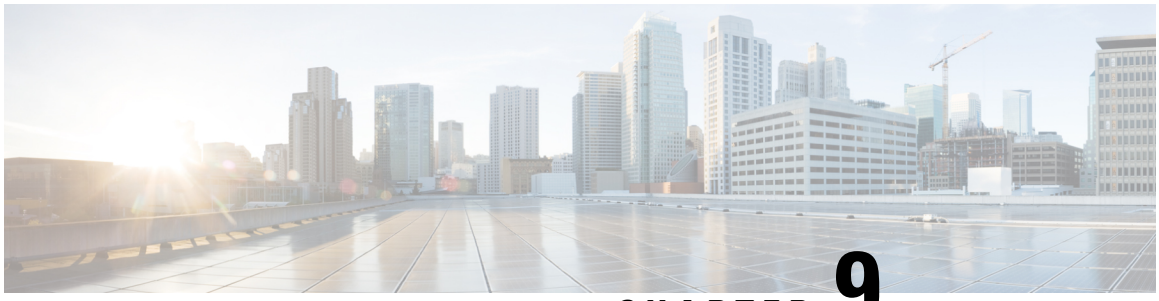
```
Device# show wireless client steering
```

Verify Cisco APs with dual-band radios

To verify the APs with dual-band radios, use the command:

```
Device# show ap dot11 dual-band summary


AP Name Subband Radio      Mac   Status Channel Power Level Slot ID Mode
-------------------------------------------------------------------------------
4800   All 3890.a5e6.f360 Enabled (40)* *1/8     (22 dBm)      0   Sensor
4800   All 3890.a5e6.f360 Enabled N/A    N/A       2              Monitor
```



CHAPTER 9

802.1x Support

- 802.1X authentication, on page 85

802.1X authentication

IEEE 802.1X port-based authentication is a network security protocol that

- prevents unauthorized devices from accessing the network
- utilizes EAP authentication models to ensure secure communication, and
- integrates with devices like routers, switches, and access points based on configuration.

Feature history

Table 2: Feature history for 802.1X authentication

Feature name	Release information	Feature description
802.1X authentication	Cisco IOS XE 16.9.1	IEEE 802.1X port-based authentication is a network security protocol that utilizes EAP authentication models to ensure secure communication, and integrates with devices like routers, switches, and access points based on configuration.
Access ports with dual port authentication	Cisco IOS XE 17.17.1	The access ports with dual port authentication feature supports dual Ethernet ports on Cisco Catalyst 9136 APs and Cisco Wireless 9178I APs.

Currently, Cisco Wave 2 and Wi-Fi 6 (802.11AX) APs support 802.1X authentication with switch ports for EAP-FAST, EAP-TLS, and EAP-PEAP methods. Configuration and credential provision to APs can be done through the controller.

Note If the AP is dot1x EAP-FAST, upon reboot, it should perform an anonymous PAC provision using ADH cipher suites to establish an authenticated tunnel. Authentication will fail if RADIUS servers do not support ADH cipher suites.

EAP-FAST protocol

In the EAP-FAST protocol developed by Cisco, to establish a secured TLS tunnel with RADIUS, the AP requires a strong shared key (PAC), provided via in-band or manual out-band provisioning.

- The EAP-FAST type configuration requires 802.1x credentials configuration for AP, since AP will use EAP-FAST with MSCHAP Version 2 method.
- Cisco 7925 phones do not support Local EAP.
- In Cisco Wave 2 APs, for 802.1x authentication using EAP-FAST after PAC provisioning (caused by the initial connection or after AP reload), ensure that you configure the switch port to trigger re-authentication using **authentication timer restart** *num* or **authentication timer reauthenticate** *num*.

EAP-TLS/EAP-PEAP Protocol

The EAP-TLS protocol or EAP-PEAP protocol provides certificate based mutual EAP authentication.

In EAP-TLS, both the server and the client side certificates are required, where the secured shared key is derived for the particular session to encrypt or decrypt data. In EAP-PEAP, only the server-side certificate is required, and the client authenticates using a password-based protocol in a secured channel.

The EAP-PEAP type configuration requires Dot1x credentials configuration for the AP, and the AP also needs to go through LSC provisioning. AP uses the PEAP protocol with MSCHAP Version 2 method.

802.1X authentication limitations

The 802.1X authentication limitations are:

- 802.1X is not supported on dynamic ports or Ethernet Channel ports.
- 802.1X is not supported in a mesh AP scenario.
- There is no recovery from the embedded controller on credential mismatch or the expiry/invalidity of the certificate on AP. The 802.1X authentication has to be disabled on the switch port to connect the AP back to fix the configurations.
- There are no certificate revocation checks implemented on the certificates installed in AP.
- You can provision only one Locally Significant Certificate (LSC) for the AP. Use the same certificate for both CAPWAP DTLS session establishment and 802.1X authentication with the switch. If you disable the global LSC configuration on the controller, the AP deletes the already provisioned LSC.
- If the AP has clear configurations applied, it will lose the 802.1X EAP Type configuration and the LSC Certificates. The AP should undergo the staging process again if 802.1X is required.
- 802.1X for trunk port APs on multi-host authentication mode is supported. Network Edge Authentication Topology (NEAT) is not supported on COS APs.

Topology

This topic explains how an AP acts as an 802.1X supplicant and how a switch uses a RADIUS server that supports EAP-FAST, EAP-TLS, and EAP-PEAP to authenticate it.

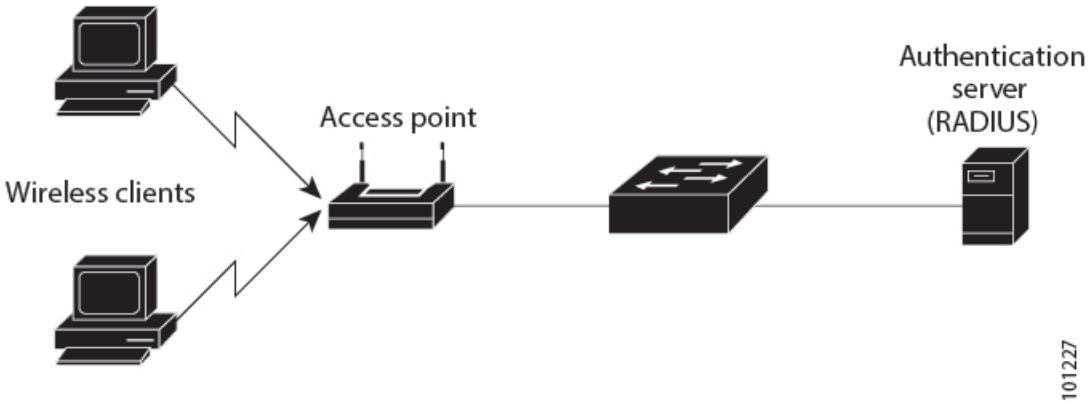
Summary

When dot1x authentication is enabled on a switch port, the connected device must authenticate itself to send and receive data other than 802.1X traffic.

- For EAP-FAST authentication, configure the RADIUS server's credentials at the controller and pass them to the AP through a configuration update request.
- For EAP-TLS or EAP-PEAP, the APs use certificates provided by the local CA server.

Workflow

Figure 3: Topology for 802.1X authentication



Configuring 802.1X authentication type and LSC AP authentication type (GUI)

Complete this task to configure 802.1X authentication type and LSC AP authentication type.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** page, click **Add**.

The **Add AP Join Profile** page is displayed.

Step 3 In the **AP** > **General** tab, navigate to **AP EAP Auth Configuration**.

Step 4 From the **EAP Type** drop-down list, choose the EAP type as **EAP-FAST**, **EAP-TLS**, or **EAP-PEAP** to configure the dot1x authentication type.

Step 5 From the **AP Authorization Type** drop-down list, choose either **CAPWAP DTLS +** or **CAPWAP DTLS**.

Step 6 Click **Save & Apply to Device**.

The 802.1X authentication type and LSC AP authentication type are configured.

Configure 802.1X authentication type and LSC AP authentication type (CLI)

Complete this task to configure 802.1X authentication type and LSC AP authentication type.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify a profile name.

Example:

```
Device(config)# ap profile ap-profile-name
```

Step 3 Configure the dot1x authentication type.

Example:

```
Device(config-ap-profile)# dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}
```

Here

- **max-sessions**: Configures the maximum 802.1X sessions initiated per AP.
- **username**: Configures the 802.1X username for all APs.
- **eap-type:**: Configures the dot1x authentication type with the switch port.
- **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.

Step 4 Configure the dot1x authentication type as EAP-FAST, EAP-TLS, or EAP-PEAP.

Example:

```
Device(config-ap-profile)# dot1x eap-type {EAP-FAST | EAP-TLS | EAP-PEAP}
```

Step 5 Configure the LSC authentication state on the AP.

Example:

```
Device(config-ap-profile)# dot1x lsc-ap-auth-state {CAPWAP-DTLS | Dot1x-port-auth | Both}
```

Here

- **CAPWAP-DTLS**: Uses LSC only for CAPWAP DTLS.
- **Dot1x-port-auth**: Uses LSC only for dot1x authentication with port.
- **Both**: Uses LSC for both CAPWAP-DTLS and Dot1x authentication with port.

Step 6 Exit the AP profile configuration mode and enter the privileged EXEC mode.

Example:

```
Device(config-ap-profile)# end
```

The 802.1X authentication type and LSC AP authentication type have been configured.

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# dot1x eap-type
Device(config-ap-profile)# dot1x eap-type EAP-TLS
Device(config-ap-profile)# dot1x lsc-ap-auth-state Dot1x-port-auth
Device(config-ap-profile)# end
```

Configure the 802.1X username and password (GUI)

Complete these steps to configure the 802.1X username and password.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join** page, click the name of the AP Join profile or click **Add** to create a new one.

Step 3 Click the **Management** tab and then click **Credentials**.

Step 4 Enter the local username, password details, and choose the appropriate local password type.

Step 5 Enter 802.1X username and password details.

Step 6 Choose the appropriate 802.1X password type.

Step 7 Enter the time in seconds after which the session should expire.

Step 8 Enable local credentials or 802.1X credentials, or both, as required.

Step 9 Click **Update & Apply to Device**.

The configuration of the 802.1X username and password is complete.

Configure the 802.1X username and password (CLI)

Complete these steps to configure the 802.1X username and password.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify a profile name for the AP.

Example:

```
Device(config)# ap profile ap-profile-name
```

Step 3 Configure the dot1x authentication type.

Example:

```
Device(config-ap-profile)# dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}
```

Here

- **max-sessions**: Configures the maximum 802.1X sessions initiated per AP.
- **username**: Configures the 802.1X username for all APs
- **eap-type:**: Configures the dot1x authentication type with the switch port.
- **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.

Step 4 Configure the dot1x password for all the APs.

Example:

```
Device(config-ap-profile)# dot1x username username password {0 | 8} password
```

Here

- **0**: Specifies an unencrypted password will follow.
- **8**: Specifies an AES encrypted password will follow.

The configuration of the 802.1X username and password is complete.

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# dot1x eap-type
Device(config-ap-profile)# dot1x username username password 0 password
```

Enable 802.1X on the switch port (CLI)

Complete these steps to enable 802.1X on the switch port.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable AAA.

Example:

```
Device(config)# aaa new-model
```

Step 3 Create authentication methods to determine user privilege for accessing the privileged command level, enabling the device to communicate with the AAA server.

Example:

```
Device(config)# aaa authentication dot1x {default | listname} method1[method2...]
```

Step 4 Enable AAA authorization for network services on 802.1X.

Example:

```
Device(config)# aaa authorization network group
```

Step 5 Enable 802.1X port-based authentication, globally.

Example:

```
Device(config)# dot1x system-auth-control
```

Step 6 Enter the interface configuration mode and specify the interface to be enabled for 802.1X authentication.

Example:

```
Device(config)# interface type slot/port
```

Step 7 Enable 802.1X port-based authentication on the interface.

Example:

```
Device(config-if)# authentication port-control {auto | force-authorized | force-unauthorized}
```

Here are the options:

- **auto**: Enables IEEE 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 Device requests the identity of the supplicant and begins relaying authentication messages between the supplicant and the authentication server. Each supplicant attempting to access the network is uniquely identified by the Device by using the supplicant MAC address.
- **force-authorized**: Disables IEEE 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 IEEE 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 supplicant to authenticate. The Device cannot provide authentication services to the supplicant through the port.

Step 8 Enable 802.1X authentication on the port with default parameters.

Example:

```
Device(config-if)# dot1x pae [supplicant | authenticator | both]
```

Step 9 Enter the privileged EXEC mode.

Example:

```
Device(config-if)# end
```

802.1X is enabled on the switch port.

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication dot1x default group radius
Device(config)# aaa authorization network group
Device(config)# dot1x system-auth-control
Device(config)# interface fastethernet2/1
Device(config-if)# authentication port-control auto
Device(config-if)# dot1x pae authenticator
Device(config-if)# end
```

Verify 802.1X on the switch port

To display the authentication state of 802.1X on the switch port, use the following command:

```
Device# show dot1x all
Sysauthcontrol              Enabled
Dot1x Protocol Version      2
Dot1x Info for FastEthernet1
-----------------------------------
PAE                         = AUTHENTICATOR
PortControl                 = AUTO
ControlDirection            = Both
HostMode                    = MULTI_HOST
ReAuthentication            = Disabled
QuietPeriod                 = 60
ServerTimeout               = 30
SuppTimeout                 = 30
ReAuthPeriod                = 3600 (Locally configured)
ReAuthMax                   = 2
MaxReq                      = 2
TxPeriod                    = 30
RateLimitPeriod             = 0
```

Verify the authentication type

To display the authentication state of an AP profile, use the following command:

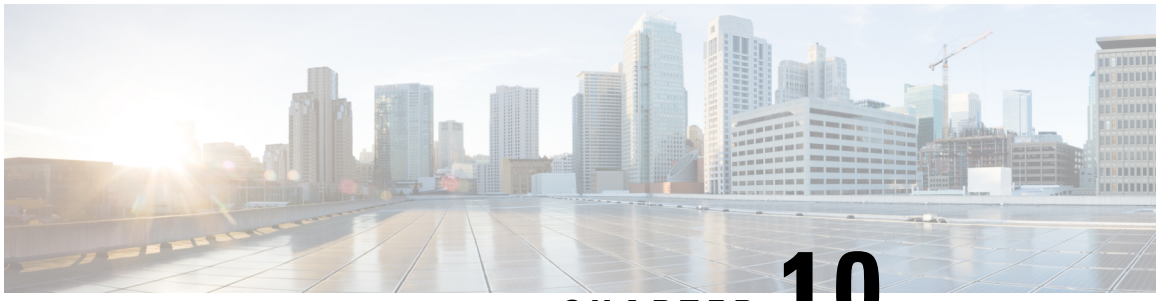
```
Device#show ap profile default-ap-profile detailed
        AP Profile Name        : default-ap-profile
        Description            : default ap profile
        …
        Dot1x EAP Method       : [EAP-FAST/EAP-TLS/EAP-PEAP/Not-Configured]
        LSC AP AUTH STATE      : [CAPWAP DTLS / DOT1x port auth / CAPWAP DTLS + DOT1x port
 auth
```



PART III

Radio Resource Management

- Radio Resource Management, on page 95
- Coverage Hole Detection, on page 115
- Cisco Flexible Radio Assignment, on page 119
- XOR Radio Support, on page 127
- Cisco Receiver Start of Packet, on page 135
- Client Limit, on page 139
- IP Theft, on page 143
- Unscheduled Automatic Power Save Delivery, on page 147
- Enabling USB Port on Access Points, on page 149



CHAPTER 10

Radio Resource Management

- Radio resource management, on page 95
- Restrictions, on page 99
- How to Configure RRM, on page 100
- Monitoring RRM Parameters and RF Group Status, on page 111
- Examples: RF group configuration, on page 112
- Event-Driven Radio Resource Management , on page 113

Radio resource management

A Radio Resource Management (RRM) is a system that

- consistently manages real-time RF parameters of a wireless network
- monitors associated APs for traffic load, interference, noise, coverage, and other metrics,
- performs critical functions like radio resource monitoring, power control transmission, dynamic channel assignment, and coverage hole detection and correction.

Feature history

Table 3: Feature history table for radio resource management

Feature name	Release information	Feature description
Radio resource management	Cisco IOS XE 16.10.1	Radio Resource Management (RRM) is a feature that automates and optimizes the management of radio frequencies in a wireless network. It enables continuous monitoring of access points (APs) for metrics such as traffic load, interference, noise, and coverage.

Functions of radio resource management

- Radio Resource Monitoring: Ensures optimal allocation of network resources.

- Power Control Transmission: Adjusts power levels to maintain network performance.
- Dynamic Channel Assignment: Allocates channels dynamically to reduce interference and optimize network performance.
- Coverage Hole Detection and Correction: Identifies and rectifies gaps in coverage to ensure consistent connectivity.
- RF Grouping: Groups RF resources effectively to manage interference and optimize performance.

Radio resource monitors

Radio resource monitor is a system that

- detects and configures new devices and APs automatically
- adjusts associated APs for optimal coverage and capacity, and
- supports noise and interference monitoring.
- APs scan all the valid channels for the country of operation as well as for channels available in other locations.
- The APs in local mode go *offchannel* for a period not greater than 70 ms to monitor these channels for noise and interference.
- Packets collected during this time are analyzed to detect rogue APs, rogue clients, ad-hoc clients, and interfering APs.
- In the presence of voice traffic or other critical traffic (in the last 100 ms), APs can defer off-channel measurements. The APs also defer off-channel measurements based on the WLAN scan priority configurations.
- Each AP spends only 0.2 percent of its time off channel. This activity is distributed across all the APs so that adjacent APs are not scanning at the same time, which could adversely affect wireless LAN performance.

Mobility controller and mobility agent

Transmit power control

A transmit power control is an automation algorithm that:

- increases and decreases an access point's power dynamically
- responds to changes in the RF coverage environment, and
- provides enough RF power to achieve the required coverage levels while avoiding channel interference.

This feature is different from coverage hole detection, which is primarily concerned with clients.

- TPC provides enough RF power to achieve the required coverage levels while avoiding channel interference between APs. We recommend that you select TPCv1; TPCv2 option is deprecated.

- With TPCv1, you can select the channel aware mode; we recommend that you select this option for 5 GHz, and leave it unchecked for 2.4 GHz.

Override the TPC algorithm with minimum and maximum transmit power settings

A Transmit Power Control (TPC) minimum and maximum transmit power setting is a wireless network configuration option that

- defines the allowed range of RF transmit powers for APs
- overrides the automatic power adjustment recommendations of the TPC algorithm, and
- applies settings globally to APs through RF profiles.

The TPC (Transmit Power Control) algorithm automatically balances RF power in various environments. Occasionally, site or architectural constraints require manually overriding TPC recommendations. With minimum and maximum power settings, you ensure APs do not exceed or fall below specific transmit powers, regardless of TPC or automatic adjustments or coverage hole detection.

Each AP model and each regulatory domain has its own allowed power levels. The increments vary: Cisco APs use 3 dB increments, but settings can be chosen in 1 dB increments and rounded.

To set the Maximum Power Level Assignment and Minimum Power Level Assignment, enter the maximum and minimum transmit power used by RRM in the fields in the **Tx Power Control** window. The range for these parameters is from –10 to 30 dBm. The minimum value cannot be greater than the maximum value; the maximum value cannot be less than the minimum value.

If you configure a maximum transmit power, RRM does not allow any access point attached to the controller, to exceed this transmit power level (whether the power is set by RRM TPC or by coverage hole detection). For example, if you configure a maximum transmit power of 11 dBm, no AP will transmit above 11 dBm unless the AP is configured manually.

Dynamic channel assignment

A dynamic channel assignment (DCA) is a wireless LAN management technique that

- automatically evaluates radio frequency (RF) conditions and network utilization
- dynamically allocates channels among APs to minimize interference and maximize performance, and
- continuously updates channel assignments based on system-wide RF analytics and policies.

Features of DCA

Features of DCA are:

- **Dynamic channel allocation**: DCA dynamically assigns channels to APs to avoid conflicts and interference, improving network capacity and performance. Two adjacent APs on the same channel can cause signal contention or collision. In a collision, data is not received by the AP. For example, reading an e-mail in a café can affect the performance of an AP in a neighboring business.

Even though these are separate networks, someone sending traffic to the café on channel 1 can disrupt communication in an enterprise using the same channel. Devices can dynamically allocate AP channel assignments to avoid conflict and increase capacity and performance.

- **Channel reuse**: Efficiently reuses channels by assigning the same channel to APs that are physically far apart, maximizing scarce RF resources. In other words, channel 1 is allocated to a different AP far from the café, which is more effective than not using channel 1 altogether.
- **Adjacent channel separation**: The device's DCA capabilities are also useful in minimizing adjacent channel interference between APs.

  For example, two overlapping channels in the 802.11b/g band, such as 1 and 2, cannot simultaneously use 11 or 54 Mbps. By effectively reassigning channels, the device keeps adjacent channels separated.

Channel assignments

The device examines a variety of real-time RF characteristics to efficiently handle channel assignments.

- AP received energy: The received signal strength measured between each AP and its nearby neighboring AP. Channels are optimized to give you the highest network capacity.
- Noise: Noise can limit signal quality for your devices and APs. Increased noise reduces cell size and degrades user experience. By optimizing channels to avoid noise sources, the device helps you maintain coverage and system capacity. If a channel is unusable due to excessive noise, that channel can be avoided.
- 802.11 interference: Interference is any 802.11 traffic that is not a part of your wireless LAN, including rogue APs and neighboring wireless networks. Lightweight APs automatically scan all channels to detect interference sources. If the amount of 802.11 interference exceeds a predefined configurable threshold (the default is 10 percent), the AP sends an alert to the device. Using the RRM algorithms, the device may then dynamically rearrange channel assignments to increase system performance in the presence of the interference. Such an adjustment could result in adjacent lightweight APs being on the same channel, but this setup provides better performance than keeping APs on a channel made unusable by interference.

  In addition, if other wireless networks are present, the device shifts the usage of channels to complement the other networks. For example, if one network is on channel 6, an adjacent wireless LAN is assigned to channel 1 or 11. This arrangement increases the capacity of the network by limiting the sharing of frequencies. If a channel has virtually no capacity remaining, the device may choose to avoid this channel. In huge deployments in which all nonoverlapping channels are occupied, the device does its best, but you must consider RF density when setting expectations.
- Load and utilization: When utilization monitoring is enabled, capacity calculations can consider that some APs are deployed in ways that carry more traffic than other APs, for example, a lobby versus an engineering area. The device can then assign channels to improve the AP that has performed the worst. The load is taken into account when changing the channel structure to minimize the impact on the clients that are currently in the wireless LAN. This metric keeps track of every AP's transmitted and received packet counts to determine how busy the APs are. New clients avoid an overloaded AP and associate to a new AP. This *Load and utilization* parameter is disabled by default.

The device combines this RF characteristic information with RRM algorithms to make system-wide decisions. Conflicting demands are resolved using soft-decision metrics that guarantee the best choice for minimizing network interference. The result is optimal channel configuration across three dimensions. APs located on different floors play an important role in your wireless LAN configuration.

RRM startup mode

The RRM startup mode is invoked under these conditions:

- In a single- device environment, the RRM startup mode is invoked after the device is upgraded and rebooted.
- In a multiple- device environment, the RRM startup mode is invoked after an RF Group leader is elected.
- You can trigger the RRM startup mode using the **ap dot11 {24ghz | 5ghz | 6ghz} rrm dca restart** command.

The RRM startup mode runs for 100 minutes (ten iterations at ten-minute intervals). The duration of the RRM startup mode is independent of the DCA interval, sensitivity, and network size. The startup mode consists of ten DCA runs with high sensitivity (making channel changes easy and sensitive to the environment) to converge to a steady-state channel plan. DCA continues to run at the specified interval and sensitivity after the startup mode is finished.

Coverage hole detection and correction

A coverage hole detection and correction algorithm is a wireless LAN management mechanism that

- identifies areas with insufficient radio coverage for reliable performance
- alerts administrators when access points fail to provide adequate coverage, and
- adjusts AP transmit power to mitigate correctable coverage holes.

If clients on a lightweight AP are detected at threshold levels such as RSSI, failed client count, percentage of failed packets, and number of failed packets that are lower than those specified in the RRM configuration, the AP sends a “coverage hole” alert to the device. The alert indicates that clients cannot connect to a usable AP because of poor signal coverage.

The device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific AP.

The device does not mitigate coverage holes caused by clients that are unable to increase their transmit power or are statically set to a power level. Increasing downstream transmit power could increase interference in the network.

Restrictions

The restrictions for RRM are:

- If an AP tries to join the RF-group that already holds the maximum number of APs it can support, the device rejects the application and throws an error.
- RRM grouping does not occur when an AP operates in a static channel that is not in the DCA channel list. The Neighbor Discovery Protocol (NDP) is sent only on DCA channels; therefore, when a radio operates on a non-DCA channel, it does not receive NDP on the channel.

How to Configure RRM

Configure neighbor discovery type (CLI)

Specify how neighbor discovery packets are handled on each radio band.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the neighbor discovery type for the desired radio band.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm ndp-type {protected | transparent}
```

The NDP types are:

- **protected**: Use **protected** to encrypt discovery packets.
- **transparent**: Use **transparent** to send packets as is (default).

Step 3 Return to privileged EXEC mode by ending the configuration mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The neighbor discovery type for the specified band is set.

Configuring Transmit Power Control

Configure Tx-power control threshold (CLI)

Set the Tx-power control threshold to define the minimum received signal strength at which the device adjusts its transmit power.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the Tx-power control threshold used by RRM for auto power assignment.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm tpc-threshold threshold_value
```

The range is from –80 dBm to –50 dBm.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

The Tx-power control threshold is updated, enabling the device to adjust its transmit power according to the specified threshold.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm tpc-threshold -60
Device(config)# end
```

Configure the Tx-power level (CLI)

Set the transmit power level of the wireless AP to improve wireless coverage and signal strength.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the 802.11 Tx-power level.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm txpower {trans_power_level | auto | max | min
| once}
```

The Tx-power parameters are:

- **trans_power_level**: Sets the transmit power level.
- **auto**: Enables auto-RF.
- **max**: Configures the maximum auto-RF Tx-power.
- **min**: Configures the minimum auto-RF Tx-power.
- **once**: Enables one-time auto-RF.

Step 3 **end**

Example:

```
Device(config)# end
```

Return to privileged EXEC mode.

Configuring 802.11 RRM Parameters

Configure 802.11 channel assignment parameters (CLI)

Configure DCA and related parameters on 802.11 radios.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure CleanAir event-driven RRM parameters.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event sensitivity {high | low
 | medium}
```

The types of sensitivity are:

- **High** : Specifies the most sensitivity to non-Wi-Fi interference as indicated by the air quality (AQ) value.
- **Low** : Specifies the least sensitivity to non-Wi-Fi interference as indicated by the AQ value.
- **Medium** : Specifies medium sensitivity to non-Wi-Fi interference as indicated by the AQ value.

Step 3 Configure DCA algorithm parameters for the 802.11 band.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel dca {add channel-number | anchor-time
| global {auto | once} | interval | min-metric | remove channel-number | sensitivity {high
 | low | medium}}
```

The DCA algorithm parameters include:

- : Enter a channel number to be added to the DCA list.
- **anchor-time**: Configures the anchor time for DCA. The range is from 0 to 23 hours.
- **global**: Configures the DCA mode for all 802.11 Cisco APs.
  - **auto**: Enables auto-RF.
  - **once**: Enables auto-RF only once.
- **interval**: Configures the DCA interval value. The values are 1, 2, 3, 4, 6, 8, 12 and 24 hours, and the default value 0 denotes 10 minutes.

- **min-metric**: Configures the DCA minimum RSSI energy metric. The range is from -100 to -60.
- **sensitivity**: Configures the DCA sensitivity level to changes in the environment.
  - **high**: Specifies the most sensitivity.
  - **low**: Specifies the least sensitivity.
  - **medium**: Specifies medium sensitivity.

Step 4 Configure the DCA channel bandwidth for all 802.11 radios in the 5-GHz band.

Example:

```
Device(config)# ap dot11 5ghz rrm channel dca chan-width {20 | 40 | 80}
```

The channel bandwidth can be set to 20 MHz, 40 MHz, or 80 MHz, . The default value for channel bandwidth is 20 MHz (80 MHz is the default value for Best). Set the channel bandwidth to Best before configuring the constraints.

The 802.11 channel assignment parameters are configured.

What to do next

Configure the advanced channel assignment parameters.

Configure the advanced channel assignment parameters (CLI)

Procedure

Step 1 Configure the persistent non-Wi-Fi device avoidance in the 802.11 channel assignment.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel device
```

Step 2 Configure the foreign AP 802.11 interference avoidance in the channel assignment.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel foreign
```

Step 3 Configure the Cisco AP 802.11 load avoidance in the channel assignment.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel load
```

Step 4 Configure noise avoidance in 802.11 channel assignment.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel noise
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

The 802.11 advanced channel assignment parameters are configured.

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel noise
  Device(config)# end
```

Configure 802.11 coverage hole detection (CLI)

Set up coverage hole detection for your wireless network.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the 802.11 coverage hole detection for data packets.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage data {fail-percentage | packet-count
| rssi-percentage} value
```

The attributes for the data packets are:

- **fail-percentage**: Configures the 802.11 coverage failure rate threshold for uplink data packets as a percentage that ranges from 1 to 100 percent.
- **packet-count**: Configures the 802.11 coverage minimum failure count threshold for uplink data packets that ranges from 1 to 255.
- **rssi-threshold**: Configures the 802.11 minimum receive coverage level for data packets that range from –90 dBm to –60 dBm.

Step 3 Configure the 802.11 AP coverage exception level as a percentage that ranges from 0 to 100 percent.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage exception global exception-level
```

Step 4 Set the minimum exception level for AP clients.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage level global cli_min exception level
```

The value range is from 1 to 75 clients.

Step 5 Configure the 802.11 coverage hole detection for voice packets.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage voice { fail-percentage |  packet-count
 | rssi-threshold} value
```

The attributes for voice packets are:

- **fail-percentage**: Configures the 802.11 coverage failure rate threshold for uplink voice packets as a percentage that ranges from 1 to 100 percent.
- **packet-count**: Configures the 802.11 coverage minimum failure count threshold for uplink voice packets that ranges from 1 to 255.
- **rssi-threshold**: Configures the 802.11 minimum receive coverage level for voice packets that range from –90 dBm to –60 dBm.

Step 6 Return to privileged EXEC mode to complete configuration.

Example:

```
Device(config)# end
```

ress **Ctrl-Z** to exit global configuration mode.

After you configure coverage hole detection thresholds, the wireless APs monitor your network and send alerts about coverage gaps.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm coverage data fail-percentage 60
Device(config)# ap dot11 6ghz rrm coverage data fail-percentage 60
Device(config)# ap dot11 24ghz rrm coverage exception global 50
Device(config)# ap dot11 24ghz rrm coverage level global 10
Device(config)# ap dot11 24ghz rrm coverage voice packet-count 10

Device(config)# end
```

Configure 802.11 event logging (CLI)

Enable and customize event logging for 802.11 wireless network parameters.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure event logging for various parameters.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm logging {channel | coverage | foreign | load |
 noise | performance | txpower}
```

The event logging parameters include:

- **channel**: Configures the 802.11 channel change logging mode.
- **coverage**: Configures the 802.11 coverage profile logging mode.
- **foreign**: Configures the 802.11 foreign interference profile logging mode.

- **load**: Configures the 802.11 load profile logging mode.
- **noise**: Configures the 802.11 noise profile logging mode.
- **performance**: Configures the 802.11 performance profile logging mode.
- **txpower**: Configures the 802.11 transmit power change logging mode.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

You have enabled 802.11 event logging for the specified parameters.

```
Device# configure terminal
Device(config)# ap dot11 {24ghz | 5ghz} rrm logging {channel | coverage | foreign | load |
 noise | performance | txpower}

Device(config)# end
```

Configure 802.11 statistics monitoring (CLI)

Enable or customize the monitoring of 802.11 statistics on wireless APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Set the 802.11 monitoring channel-list for parameters such as noise/interference/rogue.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm monitor channel-list {all | country | dca}
```

The channel list parameters include:

- **all**: Monitors all channels.
- **country**: Monitor channels used in configured country code.
- **dca**: Monitor channels used by dynamic channel assignment.

Step 3 Configure the 802.11 coverage measurement interval in seconds, which ranges from 60 to 3,600.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm monitor coverage interval
```

Step 4 Configure the 802.11 load measurement interval in seconds, which ranges from 60 to 3,600.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm monitor load interval
```

Step 5 Configure the 802.11 noise measurement interval (channel scan interval) in seconds, which ranges from 60 to 3,600.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm monitor noise interval
```

Step 6 Configure the 802.11 signal measurement interval (neighbor packet frequency) in seconds, which ranges from 60 to 3,600.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm monitor signal interval
```

Step 7 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

The system applies your 802.11 statistics monitoring settings to the APs.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm monitor channel-list all
Device(config)# ap dot11 24ghz rrm monitor coverage 600
Device(config)# ap dot11 24ghz rrm monitor load 180
Device(config)# ap dot11 24ghz rrm monitor noise 360
Device(config)# ap dot11 24ghz rrm monitor signal 480
Device(config)# end
```

Configure the 802.11 performance profile (CLI)

Configure threshold values for 802.11 performance parameters, including clients, interference, noise, throughput, and RF utilization, on Cisco APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Set the threshold value for 802.11 Cisco AP clients, which ranges from 1 to 75 clients.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm profile clients cli_threshold_value
```

Step 3 Set the threshold value for 802.11 foreign interference, which ranges from 0 to 100 percent.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm profile foreign int_threshold_value
```

Step 4 Set the threshold value for 802.11 foreign noise, which ranges from –127 to 0 dBm.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm profile noise for_noise_threshold_value
```

Step 5 Set the threshold value for 802.11 Cisco AP throughput, which ranges from 1000 to 10000000 bytes per second.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm profile throughput throughput_threshold_value
```

Step 6 Set the threshold value for 802.11 RF utilization, which ranges from 0 to 100 percent.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm profile utilization rf_util_threshold_value
```

Step 7 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The AP's performance profile is updated with your specified threshold values, enabling automated monitoring and adjustment of wireless network conditions according to configured criteria.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm profile clients 20
Device(config)# ap dot11 24ghz rrm profile foreign 50
Device(config)# ap dot11 24ghz rrm profile noise -65
Device(config)# ap dot11 24ghz rrm profile throughput 10000

Device(config)# ap dot11 24ghz rrm profile throughput 10000
Device(config)# ap dot11 24ghz rrm profile utilization 75
Device(config)# end
```

Configuring Advanced 802.11 RRM

Enable channel assignment (CLI)

To optimize radio resource allocation, assign wireless channels to 802.11 APs.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enable the 802.11 channel selection update for each AP.

Example:

```
Device# ap dot11 {24ghz | 5ghz} rrm channel-update
```

Note
After enabling the feature, the DCA algorithm assigns a token for channel assignment.

The system applies the DCA algorithm to update wireless channel assignments on all APs for the specified frequency.

```
Device# enable
Device# ap dot11 24ghz rrm channel-update
```

Restart DCA operation

To restore optimal channel allocation for wireless radios, restart the Dynamic Channel Assignment (DCA).

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Restart the DCA cycle for the 802.11 radio.

Example:

```
Device# ap dot11 {24ghz | 5ghz } rrm dca restart
```

The DCA process restarts on the wireless device.

```
Device# enable
Device# ap dot11 24ghz rrm dca restart
```

Update power assignment parameters (CLI)

Adjust the wireless transmit power settings for APs to optimize coverage and performance.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Update the 802.11 transmit power for each of the APs.

Example:

```
Device# ap dot11 {24ghz | 5ghz} rrm txpower update
```

The system updates the transmit power configuration for the specified APs.

```
Device# enable
Device# ap dot11 24ghz rrm txpower update
Device# ap dot11 6ghz rrm txpower update
```

Configuring Rogue Access Point Detection in RF Groups

Configure rogue AP detection in RF groups (CLI)

Enable detection of rogue APs within RF groups on the controllers.

Before you begin

Ensure that you configure each controller in the RF group with the same RF group name.

Note The name is used to verify the authentication IE in all beacon frames. If the controllers have different names, it results in false alarms.

Procedure

Step 1 Perform this step for each AP connected to the controller.

Example:

Step 2 Return to privileged EXEC mode to continue configuration.

Example:

```
Device(config)# end
```

Press **Ctrl-Z** to exit global configuration mode as an alternative.

Step 3 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 4 Enable rogue AP detection.

Example:

```
Device(config)# wireless wps ap-authentication
```

Step 5 Set a threshold value for rogue AP alarms.

Example:

```
Device(config)# wireless wps ap-authentication threshold value
```

The system generates an alarm when the threshold value (the number of access point frames with an invalid authentication IE) is met or exceeded within the detection period.

The threshold ranges from 1 to 255. The default value is 1. Set a higher threshold value to avoid false alarms.

Note
Enable rogue AP detection and set the threshold value on every controller in the RF group.

APs on any controller with rogue detection disabled are reported as rogues.

The system enables rogue AP detection on all APs in the RF group. Alarms are triggered when the configured threshold for invalid authentication frames is met or exceeded within the detection period.

```
Device(config)# end
Device# configure terminal
Device(config)# wireless wps ap-authentication
Device(config)# wireless wps ap-authentication threshold 50
```

Monitoring RRM Parameters and RF Group Status

Monitor RRM parameters

Provide a quick reference to the commands used for monitoring Radio Resource Management (RRM) parameters, enabling users to efficiently assess and troubleshoot wireless network performance.

Table 4: Commands for monitoring Radio Resource Management

Commands	Description
show ap dot11 24ghz channel	Displays the configuration and statistics of the 802.11b monitoring.
show ap dot11 24ghz coverage	Displays the configuration and statistics of the 802.11b coverage.
show ap dot11 24ghz group	Displays the configuration and statistics of the 802.11b grouping.
show ap dot11 24ghz logging	Displays the configuration and statistics of the 802.11b event logging.
show ap dot11 24ghz monitor	Displays the configuration and statistics of the 802.11b monitoring. nnn
show ap dot11 24ghz profile	Displays 802.11b profiling information for all APs.
show ap dot11 24ghz summary	Displays the configuration and statistics of the 802.11a APs.
show ap dot11 24ghz txpower	Displays the configuration and statistics of the 802.11b transmit power control.
show ap dot11 5ghz channel	Displays the configuration and statistics of the 802.11a channel assignment.
show ap dot11 5ghz coverage	Displays the configuration and statistics of the 802.11a coverage.

Commands	Description
show ap dot11 5ghz group	Displays the configuration and statistics of the 802.11a grouping.
show ap dot11 5ghz logging	Displays the configuration and statistics of the 802.11a event logging.
show ap dot11 5ghz monitor	Displays the configuration and statistics of the 802.11a monitoring.
show ap dot11 5ghz profile	Displays 802.11a profiling information for all APs.
show ap dot11 5ghz summary	Displays the configuration and statistics of the 802.11a APs.
show ap dot11 5ghz txpower	Displays the configuration and statistics of the 802.11a transmit power control.

Verify RF group status

This section describes the new commands for RF group status.

These commands are used to verify RF group status on the .

This table lists the commands for verifying aggressive load balancing.

Table 5: Aggressive load balancing verification commands

Command	Purpose
show ap dot11 5ghz group	Displays the controller name that is the group leader for the 802.11a RF network.
show ap dot11 24ghz group	Displays the controller name that is the group leader for the 802.11b/g RF network.

Examples: RF group configuration

These are examples of RF group name configuration.

```
Device# configure terminal
Device(config)# wireless rf-network test1
Device(config)# ap dot11 24ghz shutdown
Device(config)# end
Device# show network profile 5
```

This example demonstrates how to configure rogue AP sdetection within RF groups.

```
Device#
Device# end
Device# configure terminal
Device(config)# wireless wps ap-authentication
Device(config)# wireless wps ap-authentication threshold 50
Device(config)# end
```

Event-Driven Radio Resource Management

A Event-Driven Radio Resource Management (ED-RRM) feature is a radio frequency management solution that

- continuously monitors air quality metrics
- automatically triggers channel changes when interference exceeds a set threshold, and
- blocks affected channels for a specified duration to prevent immediate reselection.

Spontaneous interference is interference that appears suddenly on a network, perhaps jamming a channel or a range of channels completely. The Cisco CleanAir spectrum event-driven RRM feature allows you to set a threshold for air quality (AQ) that, if exceeded, triggers an immediate channel change for the affected AP.

Most RF management systems can avoid interference, but this information takes time to propagate through the system. Cisco CleanAir relies on AQ measurements to continuously evaluate the spectrum and can trigger a move within 30 seconds. For example, if an AP detects interference from a video camera, it can recover by changing channels within 30 seconds of the camera becoming active.

Configure ED-RRM on the controller (CLI)

Configure Event-Driven Radio Resource Management (ED-RRM) on the controller using CLI commands.

Trigger spectrum event-driven RRM to run when a Cisco CleanAir-enabled AP detects a significant level of interference by entering these commands.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure CleanAir driven RRM parameters for the 802.11 APs.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event
```

Step 3 Configure CleanAir driven RRM sensitivity for the 802.11 APs.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event sensitivity {low | medium
 | high | custom}
```

Default selection is Medium.

Step 4 Trigger the ED-RRM event at the set threshold value.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event custom-threshold
custom-threshold-value
```

The custom threshold range is from 1 to 99.

Step 5 Enable rogue contribution.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event rogue-contribution
```

Step 6 Configure threshold value for rogue contribution.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event rogue-contribution
duty-cycle thresholdvalue
```

The range is from 1 to 99, with 80 as the default.

Step 7 Save your changes.

Example:

```
Device# write memory
```

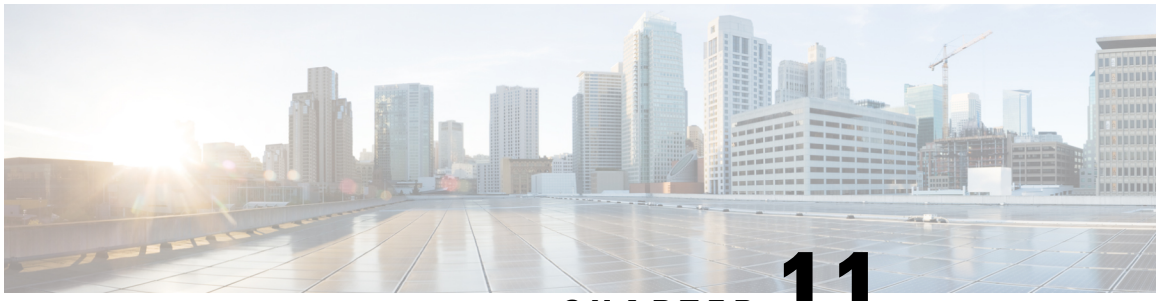
Step 8 (Optional) Verify the CleanAir configuration for the 802.11a/n/ac or 802.11b/g/n network.

Example:

```
Device# show ap dot11 {24ghz | 5ghz}cleanair config
```

The output is displayed.

The Event-Driven Radio Resource Management (ED-RRM) on the controller is configured.



CHAPTER 11

Coverage Hole Detection

• Coverage hole detection and correction, on page 115

Coverage hole detection and correction

A coverage hole detection and correction algorithm is a wireless LAN management mechanism that

- identifies areas with insufficient radio coverage for reliable performance
- alerts administrators when access points fail to provide adequate coverage, and
- adjusts AP transmit power to mitigate correctable coverage holes.

If clients on a lightweight AP are detected at threshold levels such as RSSI, failed client count, percentage of failed packets, and number of failed packets that are lower than those specified in the RRM configuration, the AP sends a “coverage hole” alert to the device. The alert indicates that clients cannot connect to a usable AP because of poor signal coverage.

The device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific AP.

The device does not mitigate coverage holes caused by clients that are unable to increase their transmit power or are statically set to a power level. Increasing downstream transmit power could increase interference in the network.

Configure coverage hole detection (GUI)

Enable Coverage hole detection (CHD) to configure client accounting using the GUI.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM**.

Step 2 Configure the Radio Resource Management parameters for 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios, and flexible radio assignment parameters.

Step 3 Check the **Enable Coverage Hole Detection** check box to activate the feature.

CHD is enabled. The system identifies and reports wireless coverage gaps.

Configure coverage hole detection (CLI)

CHD is based on upstream RSSI metrics observed by the AP. Enable CHD on your wireless device using CLI commands.



Note To revert back radios from 5-GHz to 24-GHz for CHD, ensure that the 5-GHz radio is UP and client network preference value is other than the default.

Before you begin

Disable the 802.11 network before applying the configuration.

Procedure

Step 1 Configure the 802.11 coverage level for data packets.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm coverage data {fail-percentage |
packet-count | rssi-threshold} 60
```

The options are:

- **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
- **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.
- **rssi-threshold**: Configures 802.11b minimum receive coverage level for voice packets.

Step 2 Configure the 802.11 Cisco AP coverage exception level as a percentage that ranges from zero to 100%.

Example:

```
Device(config)# ap dot11 24ghz rrm coverage exception global 50
```

Step 3 Configure the 802.11 Cisco AP client minimum exception level that ranges from one to 75 clients.

Example:

```
Device(config)# ap dot11 24ghz rrm coverage level global 10
```

Step 4 Configure the 802.11 coverage hole detection for voice packets.

Example:

```
Device(config)# ap dot11 24ghz rrm coverage voice {fail-percentage | packet-count |
rssi-threshold} 10
```

The options are:

- **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
- **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.
- **rssi-threshold**: Configures 802.11b minimum receive coverage level for voice packets.

Step 5 Save the configuration and return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 6 Verify the CHD details.

Example:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} coverage
```

This displays CHD status and statistics.

Note If both the number and percentage of failed packets exceed the values entered in the **packet-count** and **fail-percentage** commands for a five-second period, the client enters a pre-alarm condition. The controller uses this information to distinguish between real and false coverage holes. False positives typically result from poor roaming logic implemented on most clients.

A coverage hole is detected if both the number and percentage of failed clients meet or exceed the values entered in the coverage level global and coverage exception global commands over a 90-second period. The controller determines if the coverage hole can be corrected and, if appropriate, increases the transmit power level for that specific AP.

Configure CHD for RF tag profile (GUI)

Enable and configure CHD parameters for an RF tag profile using the GUI to ensure optimal wireless coverage.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM**.

Step 2 On the **Coverage** tab, select the **Enable Coverage Hole Detection** check box.

Step 3 In the **Data RSSI Threshold** field, enter the actual value in dBm. The valid value ranges from -60 dBm to -90 dBm; the default value is –80 dBm.

Step 4 In the **Voice RSSI Threshold** field, enter the actual value in dBm. The valid value ranges from -60 dBm to -90 dBm; the default value is –80 dBm.

Step 5 In the **Minimum Failed Client per AP** field, enter the minimum number of clients on an AP with a signal-to-noise ratio (SNR) below the coverage threshold. The valid value ranges from one to 75 and the default value is three.

Step 6 In the **Percent Coverage Exception Level per AP** field, enter the maximum desired percentage of clients on an AP's radio operating below the desired coverage threshold and click Apply. The valid value ranges from zero to 100% and the default value is 25%.

Step 7 In the **Voice Packet Count** and **Voice Packet Percentage** fields, enter the number and percentage of voice data packets.

Step 8 In the **Data Packet Count** and **Data Packet Percentage** fields, enter the number and percentage of data packets.

Step 9 Click **Apply**.

Configuring CHD for RF profile (CLI)

Enable CHD for a specified RF profile on your wireless device using CLI commands.

Before you begin

Ensure that the RF profile is already created.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the 802.11 coverage hole detection for data packets.

Example:

```
Device(config)# ap dot11 24ghz rf-profile alpha-rfprofile-24ghz
```

Step 3 Configure the minimum RSSI value for data packets received by the AP. The valid values range is from -90 to -60 in dBm.

Example:

```
Device(config-rf-profile)# coverage data rssi threshold -80
```

Step 4 Save the configuration and return to privileged EXEC mode.

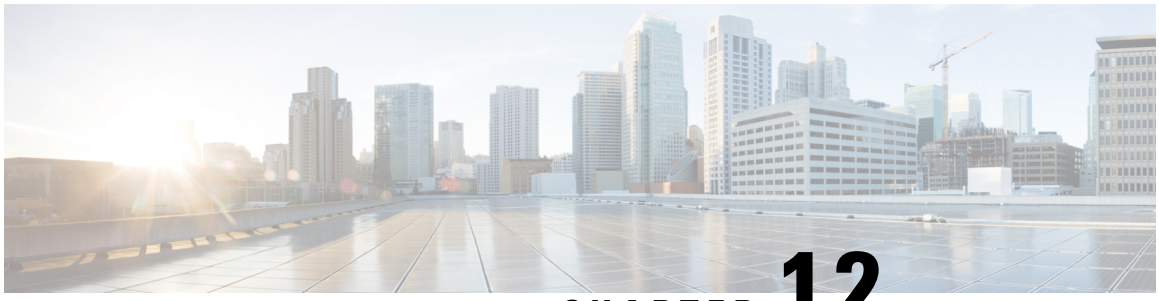
Example:

```
Device(config-rf-profile)# end
```

Step 5 Verify the summary of the available RF profiles.

Example:

```
Device# show ap dot11 24ghz rf-profile summary
```



CHAPTER 12

Cisco Flexible Radio Assignment

- Flexible radio assignments, on page 119
- Flexible radio assignment actions, on page 123

Flexible radio assignments

A Flexible Radio Assignment (FRA) is a configuration management feature that

- adapts AP radio hardware for multiple roles
- manages various Cisco AP models like Cisco Aironet 2800 Series Access Points, Cisco Aironet 3800 Series Access Points, and other APs in Catalyst series, and
- optimizes client experience by adapting radio roles based on client capabilities

FRA takes advantage of the dual-band radios included in APs. The FRA is a new feature added to the RRM to analyze the Neighbor Discovery Protocol (NDP) measurements, which manages the hardware used to determine the role of the new flexible radio (2.4 GHz, 5 GHz, or monitor) in your network.

Traditional legacy dual–band APs always had 2 radio slots, (1 slot per band) and were organized by the band they were serving, that is slot 0= 802.11b,g,n and slot 1=802.11a,n,ac.

Benefits of the FRA

- Solves the problem of 2.4–GHz over coverage.
- Creates two distinct 5–GHz cells to double available airtime.
- Permits one AP with one Ethernet drop to function like two 5–GHz APs.
- Introduces the concept of macro and micro cells for airtime efficiency.
- Allows more bandwidth to be applied to an area within a larger coverage cell.
- Addresses nonlinear traffic distribution.
- Enhances the High-Density Experience (HDX) with one AP.
- Allows XOR radios to serve client or monitor roles based on configuration.

XOR support in 2.4-GHz or 5-GHz bands

XOR support increases flexibility in AP operations by offering capabilities to serve various frequency bands and monitor radio signals, thereby optimizing network coverage and efficiency.

The flexible radio (XOR) offers the ability to serve the 2.4-GHz or the 5-GHz bands, or passively monitor both bands on the same AP.

- AP models supporting dual 5-GHz band operations:
  - *i* model supports a dedicated Macro/Micro architecture.
  - *e* and *p* models support Macro/Macro architecture.
- FRA with internal antenna (*i* series models) allows two 5-GHz radios in Micro/Macro cell mode.
- FRA with external antenna (*e* and *p* models) enables creation of two separate macro or micro cells for HDX.
- FRA calculates redundancy for 2.4-GHz radios with a metric called COF (Coverage Overlap Factor).
- Feature integration in RRM for mixed environments and AP **AP MODE** selections include:
  - Local Mode
  - Monitor Mode
  - FlexConnect Mode
  - Sniffer Mode
  - Spectrum Connect Mode

Before XOR was introduced, mode changes affected the entire AP (both radio slot 0 and slot 1). The XOR addition allows operation of a single radio interface independently, known as *roles*:

- Client Serving
- Either 2.4 GHz(1) or 5 GHz(2)
- Monitor-Monitor mode (3)

Note

- Mode: Assigned to a whole AP (slot 0 and slot 1)
- Role: Assigned to a single radio interface (slot 0)

Configure FRA Radio (CLI)

Procedure

Step 1 Enable privileged EXEC mode

Example:

```
Device# enable
```

Enters privileged EXEC mode.

Step 2 Enter configuration mode

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 3 Enable or disable FRA

Example:

```
Device(config)# [no] ap fra
```

Enables or disables FRA on the AP.

Step 4 Configure FRA interval

Example:

```
Device(config)# ap fra interval 3
```

Configures the FRA interval in hours. The range is 1 to 24 hours.

Note
The FRA interval has to be more than the configured RRM interval.

Step 5 Configure the FRA sensitivity

Example:

```
Device(config)# ap fra sensitivity high
```

Configures the FRA sensitivity.

- **high**: Sets the FRA Coverage Overlap Sensitivity to **high**.
- **medium**: Sets the FRA Coverage Overlap Sensitivity to **medium**.
- **low**: Sets the FRA Coverage Overlap Sensitivity to **low**.

Step 6 Exit global configuration mode

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 7 Reverts to XOR radio state

Example:

```
Device# ap fra revert all auto
```

Rolls back the XOR Radio state.

- **all**: Reverts all XOR Radios
- **auto-only**: Revert only XOR radios currently in automatic band selection.

- **auto**: Sets the XOR radios in automatic band selection.
- **static**: Sets the XOR radio in static 2.4-GHz band.

Step 8 View the configuration and statistics of 802.11 APs

```
Device# show ap dot11 5ghz summary
```

Shows the configuration and statistics of 802.11 Cisco APs

Step 9 View the current FRA configuration

Example:

```
Device# show ap fra

FRA State                                          : Disabled
FRA Sensitivity                                    : medium (95%)
FRA Interval                                       : 1 Hour(s)

AP Name              MAC Address       Slot ID  Current-Band      COF %      Suggested Mode
-----------------------------------------------------------------------------------------------
AP00A6.CA36.295A     006b.f09c.8290    0        2.4GHz            None       2.4GHz

COF : Coverage Overlap Factor

test_machine#
```

Shows the current FRA configuration.

Step 10 View the current 802.11 dual-band parameters

Example:

```
Device# show ap name config dot11 dual-band
```

Shows the current 802.11 dual-band parameters in a given AP.

Configure FRA radio (GUI)

The purpose of configuring the FRA radio is to optimize the radio assignment for overlapping coverage areas, improving network efficiency and performance. This includes enabling the Flexible Radio Assignment (FRA) status, setting intervals, and adjusting sensitivity levels to ensure optimal coverage.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM** > **FRA**.

Step 2 In the **Flexible Radio Assignment** window, enable FRA status and determine the overlapping 2.4 GHz or 5 GHz coverage for each AP, choose **Enabled** in the **FRA Status** field. By default, the FRA status is disabled.

Step 3 Under the From the **FRA Interval** drop-down list, choose the FRA run interval. The interval values range from 1 hour to 24 hours. Choose the FRA run interval value only after you enable the FRA status.

Step 4 From the **FRA Sensitivity** drop-down list, choose the percentage of Coverage Overlap Factor (COF) required to consider a radio as redundant. You can select the supported value only after you enable the FRA status.

The supported values are as follows:

- Low: 100 percent
- Medium (default): 95 percent
- High: 90 percent

The **Last Run** and **Last Run Time** fields will show the time FRA was run last and the time it was run.

Step 5 Check the **Client Aware** check box to take decisions on redundancy.

When enabled, the **Client Aware** feature monitors the dedicated 5-GHz radio and when the client load passes a pre-set threshold, automatically changes the Flexible Radio assignment from a monitor role into a 5-GHz role, effectively doubling the capacity of the cell on demand. Once the capacity crisis is over and Wi-Fi load returns to normal, the radios resume their previous roles.

Step 6 In the **Client Select** field, enter a value for client selection. The valid values range between 0 and 100 percent. The default value is 50 percent.

This means that if the dedicated 5-GHz interface reaches 50% channel utilization, this will trigger the monitor role dual-band interface to transition to a 5-GHz client-serving role.

Step 7 In the **Client Reset** field, enter a reset value for the client. The valid values range between 0 and 100 percent. The default value is 5 percent.

Once the AP is operating as a dual 5-GHz AP, this setting indicates the reduction in the combined radios' overall channel utilization required to reset the dual-band radio to monitor role.

Step 8 Click **Apply** to save the configuration.

After completing the configuration, the FRA system will be active, improving the radio coverage efficiency by managing overlapping frequencies and enhancing redundancy decisions. This results in better utilization of network resources and coverage optimization.

Flexible radio assignment actions

Flexible Radio Assignment (FRA) evaluates 2.4-GHz radio coverage to identify overlapping coverage causing radio interference. If there is an overlapping coverage, the dual-band radio moves to either 5-GHz client serving or monitor role.

Feature history for flexible radio assignment action

Table 6: Feature history for FRA Action

Release	Feature	Feature information
Cisco IOS XE Dublin 17.10.1	Flexible Radio Assignment (FRA) Action	In Cisco IOS-XE 17.10.1 and earlier releases, the FRA moves the redundant dual-band radios to either 5-GHz client-serving role or monitor role.

Release	Feature	Feature information
Cisco IOS XE Dublin17.11.1	Flexible Radio Assignment (FRA) Action	From Cisco IOS-XE 17.11.1 onwards, you can select the redundant dual-band radios in a network to operate in monitor only mode.

Release Information

In Cisco IOS-XE 17.10.1 and earlier releases, the FRA moves the redundant dual-band radios to either 5-GHz client-serving role or monitor role. From Cisco IOS-XE 17.11.1 onwards, you can set redundant dual-band radios in a network to operate in monitor-only mode.



Note The FRA action feature is disabled by default.

Configure FRA action in default RF profile (CLI)

Configure the FRA action in the default RF profile to optimize radio frequency management.

Procedure

Step 1 Configure terminal

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the FRA action

Example:

```
Device(config)# ap dot11 24ghz fra action monitor
```

Configures the FRA action to monitor mode, moving all redundant dual-band APs solely to the monitor role.

Step 3 Exit configuration mode

Example:

```
Device(config)# end
```

Exits configuration mode and returns to privileged EXEC mode.

Upon completion, the FRA action is set, redundancy in dual-band radios is handled efficiently, and the system returns to privileged EXEC mode.

Configure FRA action in 2.4-GHz RF profile (CLI)

Procedure

Step 1 Configure terminal

Example:

```
Device# configure terminal
```

Enters global configuration mode

Step 2 Configure the RF profile name

Example:

```
Device(config)# ap dot11 24ghz rf-profile alpha-rfprofile-24ghz
```

Configures the RF profile name and enters RF profile configuration mode.

Step 3 Configure the FRA action as monitor

Example:

```
Device(config-rf-profile)# fra action monitor
```

Configures the FRA action as monitor, and moves all redundant dual-band radios to monitor role only.

Step 4 Exit configuration mode

Example:

```
Device(config-rf-profile)# end
```

Exits configuration mode and returns to privileged EXEC mode.

Verify FRA action configuration

To view the selected FRA action, use the following command:

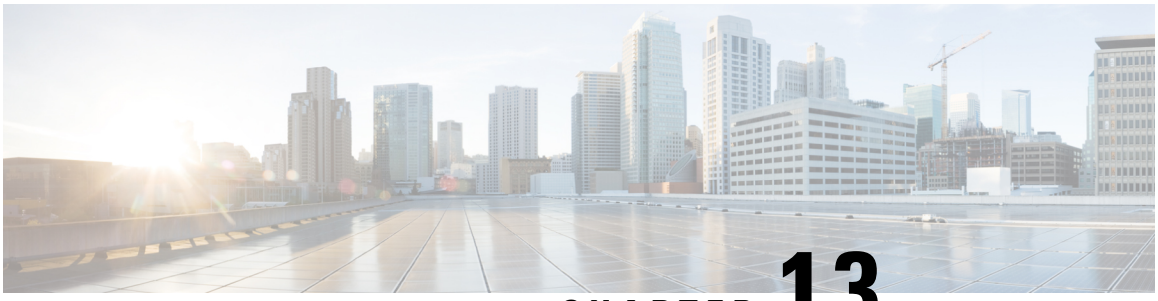
```
Device# show ap fra
FRA State              : Enabled
FRA Freeze             : Disabled
FRA Operation State    : Up
FRA Sensitivity        : higher (85%)
FRA Interval           : 1 Hour(s)
Service Priority       : Coverage
Client Aware FRA       : Enabled
  Client Select        : 25%
  Client Reset         : 5%
FRA Action             : 2.4GHz/Monitor
  Last Run             : 3069 seconds ago
```

To view the FRA action details in an AP RF profile, use the following command:

```
Device# show ap rf-profile name madhu-rf-profile-24 detail | sec FRA
Client Aware FRA                 : Disabled
FRA Action                       : 2.4GHz/Monitor
```

To view the radio mode and role in an AP, use the following command:

```
Device# show ap name AP7872.5DED.CB74 config slot 0 | sec Attribute
Attributes for Slot 0
  Radio Type                                    : 802.11n - 2.4/5 GHz
  Radio Mode                                    : Monitor
  Radio Role                                    : Monitor
    Assignment Method                           : Auto
    Monitor Mode Reason                         : Automatically Switched by FRA
```



CHAPTER 13

XOR Radio Support

- Dual-band radios in Cisco AP models, on page 127

Dual-band radios in Cisco AP models

A dual-band radio is a device category that

- offers connectivity on more than one frequency band (such as 2.4 GHz and 5 GHz)
- provides flexibility in network configuration, and
- is used in multiple Cisco AP models like the 2800, 3800, 4800, and 9120 series.

Feature history table

Table 7: Feature history table for dual-band radios in Cisco AP models

Feature name	Release information	Feature description
Dual-band (XOR) radio support	Cisco IOS XE 16.10.1	The Dual-band (XOR) radio support offers the ability to serve 2.4 GHz or 5 GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Key features of dual-band radios in Cisco APs

The XOR radio support can be steered manually or automatically.

- Manual steering of a band on a radio: The band on the XOR radio can only be changed manually.
- Automatic client and band steering on radios: This is managed by the Flexible Radio Assignment (FRA) feature that monitors and changes the band configurations as per site requirements.

Client steering

When a radio moves between bands (from 2.4 GHz to 5 GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5 GHz band, client steering algorithms contained in the FRA algorithms are used to steer a client between the same band co-resident radios

Cisco APs and dual-band radios

Cisco 2800, 3800, 4800, and 9120 series AP models are equipped with dual-band (XOR) radios. These models have

- radios that operate on either 2.4 GHz or 5 GHz bands, or
- the ability to passively monitor both the bands on the same AP.

These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Cisco AP models up to the Cisco 9120 APs are designed to support dual 5 GHz band operations with the *i* model supporting a dedicated Macro or Micro architecture and the *e* and *p* models supporting Macro or Macro. The Cisco 9130AXI APs support dual 5 GHz operations as Macro or Micro cell.

Limitations

These are the limitations for XOR radio support:

- RF measurement is disabled when a static channel is configured on slot 1. As a result, the dual-band radio slot 0 operates only with 5 GHz radios and not in the monitor mode. When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4 GHz radio.
- Only one of the 5 GHz radios can operate in the UNII band (100 to 144), due to an AP limitation to maintain the power budget within the regulatory limit.

Configure default XOR radio support (CLI)

Complete this task to configure the basic settings for XOR radio support.

Before you begin

Note The default radio points to the XOR radio hosted on slot 0.

Procedure

Step 1 Enable the priviledged EXEC mode.

Example:

```
Device> enable
```

Step 2 Configure the 802.11 dual-band external antenna gain on the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain antenna_gain_value
```

Enter external antenna gain value in multiple of .5 dBi units (that is, an integer value 4 means 4 x 0.5 = 2 dBi of gain).

Step 3 Shut down the default dual-band radio.

Example:

```
Device# ap name ap-name  dot11 dual-band shutdown
```

Use the **no** form of the command to enable the radio.

Step 4 Switch to the client-serving mode of the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band role manual client-serving
```

Step 5 Switch to the 2.4 GHz radio band.

Example:

```
Device# ap name ap-name  dot11 dual-band band 24ghz
```

The basic configuration of the XOR radio is completed.

Example

```
Device> enable
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain 2
Device# ap name ap-name  dot11 dual-band shutdown
Device# Device# ap name ap-name  dot11 dual-band role manual client-serving
Device# Device# ap name ap-name  dot11 dual-band band 24ghz
```

What to do next

Continue with the configuration steps listed in the "Configure advanced settings" to complete the configuration.

Configure advanced settings (CLI)

Complete this task to configure the advanced settings for XOR radio support, such as, transmit power, channel, channel width, CleanAir settings and so on.

Procedure

Step 1 Configure the transmit power for the radio on the AP.

Example:

```
Device# ap name ap-name  dot11 dual-band txpower {transmit_power_level | auto}
```

Note

When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio.

If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.

Step 2 Configure the channel number for the dual band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel 40
```

The range of the *channel-number* is from 1 to 173.

Step 3 (Optional) Enable the auto channel assignment for the dual-band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel auto
```

Step 4 Configure the channel width for the dual band.

Example:

```
Device# ap name ap-name  dot11 dual-band channel width {20 MHz | 40 MHz | 80 MHz | 160 MHz}
```

Step 5 Enable the Cisco CleanAir management on the dual-band radio.

Example:

```
Device# ap name ap-name  dot11 dual-band cleanair
```

Step 6 Configure a band for the Cisco CleanAir feature.

Example:

```
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
```

Selects a band for the Cisco CleanAir feature.

Use the **no** form of this command to disable the Cisco CleanAir feature.

Step 7 Configure the 802.11n dual-band antenna and antenna ports selection for a specific AP.

Example:

```
Device# ap name ap-name  dot11 dual-band dot11n antenna {A | B | C | D}
```

Here, A, B, C, and D are the antenna ports.

The configuration of the additional features for dual-band (XOR) radios are completed.

Example

```
Device# ap name ap-name  dot11 dual-band txpower auto
Device# ap name ap-name  dot11 dual-band channel 40
Device# ap name ap-name  dot11 dual-band channel width 40 MHz
Device# ap name ap-name  dot11 dual-band cleanair
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
Device# ap name ap-name  dot11 dual-band dot11n antenna A
```

Configure XOR radio support for the specified slot number (GUI)

Complete this task to configure XOR radio for the specified slot number.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, it displays the antenna PID and antenna design specifics.

Step 3 Click **Configure**.

Step 4 In the **General** tab, set the **Admin Status** as required.

Step 5 Set the **CleanAir Admin Status** field to Enable or Disable.

Step 6 Click **Update & Apply to Device**.

The XOR radio support for the specified slot number has been configured.

Configure XOR radio support for the specified slot number (CLI)

Complete this task to configure XOR radio for a specified slot number.

Procedure

Step 1 Enable the priviledged EXEC mode.

Example:

```
Device> enable
```

Step 2 Configure dual-band antenna for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 antenna ext-ant-gain
external_antenna_gain_value
```

Here, *external_antenna_gain_value* is the external antenna gain value in multiples of .5 dBi unit. The range is from 0 to 40.

Note

- For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration.
- For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.

Step 3 Configure the current band for XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 band {24ghz | 5ghz}
```

Step 4 Configures dual-band channel for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 channel {channel-number | auto | width [160
 | 20 | 40 | 80]}
```

The range of the *channel_number* is from 1 to 165.

Step 5 Enable the Cisco CleanAir feature for dual-band radios hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 cleanair band {24Ghz | 5ghz}
```

Step 6 Configure 802.11n dual-band parameters hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 dot11n antenna {A | B | C | D}
```

Here are the antenna configurations.

A : Enables antenna port A.

B : Enables antenna port B.

C : Enables antenna port C.

D : Enables antenna port D.

Step 7 Configure dual-band role for the XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 role {auto | manual [client-serving | monitor]}
```

Here are the dual-band roles.

- **auto** : refers to the automatic radio role selection.
- **manual** : refers to the manual radio role selection.

Note
When Flexible Radio Assignment (FRA) is disabled, XOR radios whose roles were changed by Radio Resource Management (RRM) might not automatically revert to their default radio role. To restore an individual AP to the default client-serving role, use the these commands:

```
Device# ap name ap-name dot11 dual-band slot 0 shutdown

Device# ap name ap-name dot11 dual-band slot 0 role manual client-serving

Device# ap name ap-name dot11 dual-band slot 0 band 24ghz

Device# ap name ap-name no dot11 dual-band slot 0 shutdown
```

Step 8 Disable dual-band radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name [no] dot11 dual-band slot 0 shutdown
```

Use the **no** form of this command to enable the dual-band radio.

Step 9 Configure dual-band transmit power for XOR radio hosted on slot 0 for a specific AP.

Example:

```
Device# ap name ap-name dot11 dual-band slot 0 txpower {tx-power-level | auto}
```

Here,

- *tx_power_level* : is the transmit power level in dBm. The range is from 1 to 8.
- **auto** : enables auto-RF.

The XOR radio for the specified slot number is configured.

Example

```
Device> enable
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna ext-ant-gain 2
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair band 24Ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n antenna A
                Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2
```

Verify XOR radio support

To verify the XOR radio support parameters, use these **show** commands:

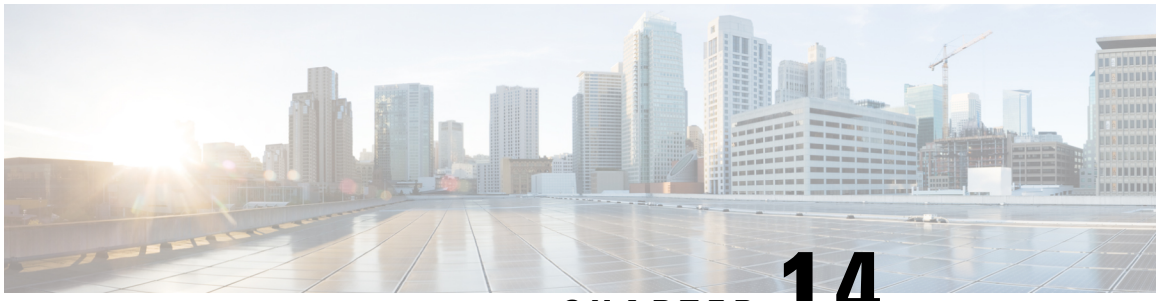
Verify auto-RF information

To display the auto-RF information for a Cisco AP, use the **show ap name** *ap-name* **auto-rf dot11 dual-band** command.

```
Device# show ap name cisco-ap-name auto-rf dot11 dual-band
```

Verify the list of BSSIDs

To display the list of BSSIDs for a Cisco AP, use the **show ap name** *ap-name* **wlan dot11 dual-band** command.



CHAPTER 14

Cisco Receiver Start of Packet

- Receiver start of packet detection threshold, on page 135

Receiver start of packet detection threshold

A receiver start of packet (Rx SOP) detection threshold is a wireless radio configuration setting that

- determines the Wi-Fi signal level (measured in dBm) at which an APs radio demodulates and decodes a received packet
- affects radio sensitivity and the effective size of the receiver cell, and
- influences how clients are distributed across APs within a wireless network.

Rx SOP is used to address clients with weak RF links, sticky clients, and to support client load balancing across APs. Rx SOP helps optimize network performance in high-density deployments, such as stadiums and auditoriums. In these environments, APs need to prioritize the nearest and strongest clients.

Restrictions for Rx SOP

- Rx SOP configuration does not apply to the third radio module pluggable on Cisco Aironet Series APs.
- Rx SOP configurations are supported only in Local, FlexConnect, Bridge, and Flex+Bridge modes.
- Rx SOP configurations are not supported in the FlexConnect+PPPoE, FlexConnect+PPPoE-wIPS, and FlexConnect+OEAP submodes.

Permitted range for the Rx SOP threshold

This table shows the allowed Rx SOP (Receiver Start of Packet) threshold settings.

- It is for different radio bands and threshold levels.

Table 8: Rx SOP Threshold

Radio Band	Threshold High	Threshold Medium	Threshold Low
2.4 GHz	-79 dBm	-82 dBm	-85 dBm

Radio Band	Threshold High	Threshold Medium	Threshold Low
5 GHz	-76 dBm	-78 dBm	-80 dBm

Known behavior

Regardless of the radio mode, the controller sets the radio with the specified RX-SOP value. The AP then decides whether to use this value.

- For the XOR radio (Slot 0), when the AP is in monitor mode, the RX-SOP value sent to the AP depends on the frequency band used before switching to monitor mode. If the radio operated in the 2.4 GHz band, the RX-SOP parameters are selected from the 2.4 GHz RF profile or the default RF profile. If it operated in the 5 GHz band, the RX-SOP parameters are chosen from the 5 GHz RF profile or the default RF profile configured for the AP.

Configure Rx SOP (CLI)

Adjust the Rx SOP threshold for high-density Wi-Fi environments.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
DeviceDevice# configure terminal
```

Step 2 Configure the 802.11bg or 802.11a radio Rx SOP threshold.

Example:

```
DeviceDevice(config)# ap dot11 {24ghz | 5ghz} rx-sop threshold {auto | custom | high | low
 | medium}
```

Step 3 Return to the privileged EXEC mode.

Example:

```
Device# end
```

Step 4 Display the 802.11bg or 802.11a high-density parameters.

Example:

```
DeviceDevice# show ap dot11 {24ghz | 5ghz} high-density
```

Step 5 Display the summary of all the connected Cisco APs.

Example:

```
DeviceDevice# show ap summary
```

Customize RF profile (CLI)

Customize 802.11 RF profile parameters.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
DeviceDevice# configure terminal
```

Step 2 Configure the 802.11a and 11b parameters.

Example:

```
DeviceDevice(config)# ap dot11 {24ghz | 5ghz | 6ghz} rf-profile profile-name AHS_2.4ghz
```

Step 3 **Example:**

```
DeviceDevice(config)# ap dot11 {24ghz | 5ghz | 6ghz} rf-profile profile-name AHS_2.4ghz
```

Step 4 Configure the 802.11bg, 802.11a high-density parameters.

Example:

```
DeviceDevice(config-rf-profile)# high-density rx-sop threshold {auto | custom | high | low
 | medium}
```

Step 5 Display the summary of all the connected Cisco APs.

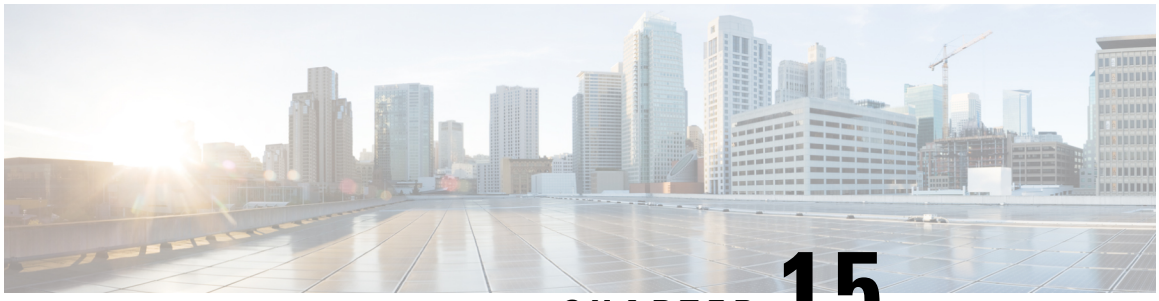
Example:

```
DeviceDevice# show ap summary
```

Step 6 Return to the privileged EXEC mode.

Example:

```
Device# end
```



CHAPTER 15

Client Limit

- Client limits, on page 139

Client limits

A client limit is a wireless network feature that

- enforces a maximum number of client devices that can connect to an AP, and
- allows you to configure the client cap per AP radio or per WLAN.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 9: Feature history for client limits

Feature Name	Release Information	Feature Description
Client Limit and Delete	Cisco IOS XE Dublin 17.8.x	From this release, client limiting is supported per AP, per radio, and per AP-radio per WLAN. Client limiting is supported on the Cisco Catalyst 9136 Series APs in FlexConnect mode. The following commands are introduced: • association-limit • high-density clients count
Client limits	Cisco IOS XE Gibraltar 16.10.x	This feature enforces a maximum number of client devices per AP, supporting per-AP, per-radio, and per-WLAN configurations.

Configure client limit per WLAN (GUI)

Restrict the number of client devices that can connect to a specific WLAN.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click a WLAN from the list of WLANs.

Step 3 Click the **Advanced** tab.

Step 4 Under the **Max Client Connections** settings, enter the client limit for **Per WLAN**, **Per AP Per WLAN**, and **Per AP Radio Per WLAN**.

Step 5 Click **Update & Apply to Device**.

Configure client limit per WLAN (CLI)

Set the maximum number of clients permitted to associate with a specific WLAN, AP, or AP radio using CLI commands.

Procedure

Step 1 Enter the privileged EXEC and the global configuration mode.

Example:

```
Device# enable
Device# configure terminal
```

Step 2 Specify the WLAN name.

Example:

```
Device(config)# wlan <wlan-profile-name>
```

WLAN needs to be shut down in order to change the client limit parameters.

Step 3 Configure the maximum number of clients that is associated to the given WLAN.

Example:

```
Device(config-wlan)# client association limit <maximum-client-per-WLAN>
```

Note
Depending on the primary AP in the Cisco Embedded Wireless Controller network, the maximum number of clients supported varies.

Table 10: Scale Supported in a Cisco Embedded Wireless Controller Network

Step 4 Return to the privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

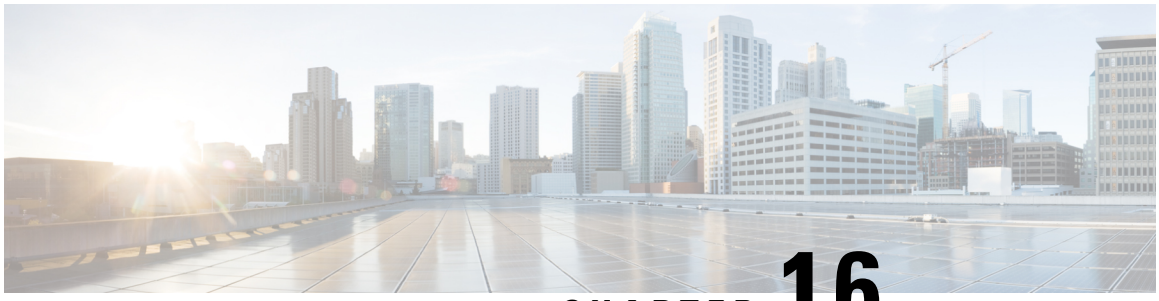
Example:

```
Device(config)# end
```

Step 5 Verify the current configuration of the WLAN and the corresponding client association limits.

Example:

```
Device# show wlan id wlan-id 2
```



CHAPTER 16

IP Theft

- IP theft, on page 143

IP theft

An IP Theft feature is a wireless controller security mechanism that

- detects duplicate IP address usage among connected clients
- assigns precedence to clients based on a defined preference order, and
- blocks or excludes clients attempting to use IP addresses already assigned to others.

The IP Theft feature is enabled by default on the controller. If a wireless client tries to use an IP address assigned to a wired client, the controller marks it as a theft attempt.

Lower the idle timeout to prevent false IP theft

Lower the idle timeout for devices that may switch WLANs while out of range to prevent incorrect IP theft events.

When a device moves between WLANs while out of range of APs, it may retain the same IPv6 link-local address but present a different MAC address. Because the controller cannot immediately detect the move, it may interpret the reuse of the same IP address as an IP theft event.

To avoid this, lower the idle timeout value so the controller can promptly remove stale client entries from the original WLAN. This helps ensure that address reuse by legitimate roaming devices is not mistakenly treated as a theft attempt.

IP preference levels for IP theft detection

The controller uses IP preference levels to resolve conflicts when multiple clients claim the same IP address. These levels determine which client has priority based on the method used to learn the IP address. Preference order for IPv4 clients:

- DHCPv4
- ARP
- Data packets

Preference order for IPv6 clients:

- DHCPv6
- NDP (Neighbor Discovery Protocol)
- Data packets

Additional rules:

- Static wired clients always receive higher preference than dynamically assigned clients.
- Wired clients are prioritized over wireless clients when IP conflicts occur.

Configure IP theft (GUI)

Prevent unauthorized use or reuse of IP addresses by configuring IP theft protections.

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies** > **Client Exclusion Policies**.

Step 2 Check the **IP Theft or IP Reuse** check box.

Step 3 Click **Apply**.

Configure IP theft (CLI)

Enable IP theft detection and configure client exclusion policies using CLI.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the client exclusion policy.

Example:

```
Device(config)# wireless wps client-exclusion ip-theft
```

Configure the IP theft exclusion timer (CLI)

Set the exclusion timer to temporarily block IP addresses suspected of theft on a WLAN.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a WLAN policy profile and enter the wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy profile-policy default-policy-profile
```

Step 3 Specify the timeout, in seconds.

Example:

```
Device(config-wireless-policy)# exclusionlist timeout time-in-seconds
```

The valid range is from zero-2147483647. Enter zero for no timeout.

Verify IP theft configuration

Use the command to check if the IP theft feature is enabled or not:

```
Device# show wireless wps summary

Client Exclusion Policy
  Excessive 802.11-association failures   : Enabled
  Excessive 802.11-authentication failures: Enabled
  Excessive 802.1x-authentication         : Enabled
  IP-theft                                : Enabled
  Excessive Web authentication failure    : Enabled
  Cids Shun failure                       : Enabled
  Misconfiguration failure                : Enabled
  Failed Qos Policy                       : Enabled
  Failed Epm                              : Enabled
```

Use the commands to view additional details about the IP theft feature:

```
Device# show wireless client summary

Number of Local Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method     Role
-------------------------------------------------------------------------------------------
000b.bbb1.0001 SimAP-1                2    Run                11a      None       Local


Number of Excluded Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method
-------------------------------------------------------------------------------------------
10da.4320.cce9 charlie2               2    Excluded           11ac     None
```

```
Device# show wireless device-tracking database ip

IP                              VLAN  STATE       DISCOVERY   MAC
  ------------------------------------------------------------------------
  20.20.20.2                     20    Reachable   Local       001e.14cc.cbff
  20.20.20.6                     20    Reachable   IPv4 DHCP   000b.bbb1.0001

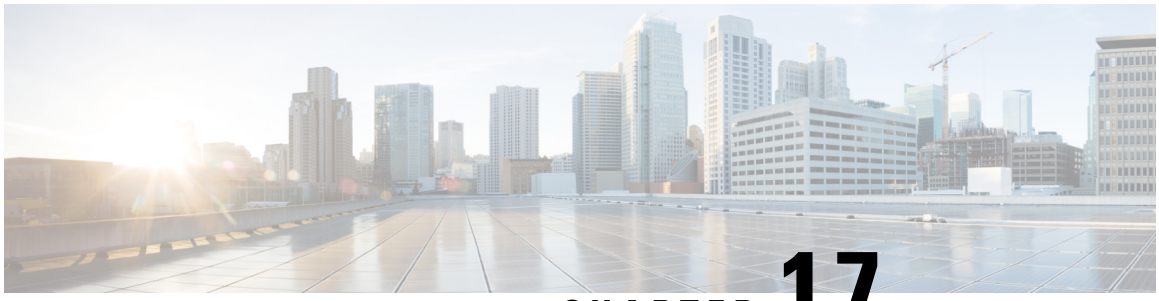

Device# show wireless exclusionlist

Excluded Clients

MAC Address       Description          Exclusion Reason               Time Remaining
-------------------------------------------------------------------------------------
10da.4320.cce9                         IP address theft                    59


Device# show wireless exclusionlist client mac 12da.4820.cce9 detail

Client State : Excluded
Client MAC Address : 12da.4820.cce9
Client IPv4 Address: 20.20.20.6
Client IPv6 Address: N/A
Client Username: N/A
Exclusion Reason : IP address theft
Authentication Method : None
Protocol: 802.11ac
AP MAC Address : 58ac.780e.08f0
AP Name: charlie2
AP slot : 1
Wireless LAN Id : 2
Wireless LAN Name: mhe-ewlc
VLAN Id : 20
```



CHAPTER 17

Unscheduled Automatic Power Save Delivery

• Unscheduled automatic power save delivery overview, on page 147

Unscheduled automatic power save delivery overview

Unscheduled automatic power save delivery (U-APSD) is a power-saving mechanism that

- is defined as a QoS facility in IEEE 802.11e standards
- delivers multiple downlink packets in response to a single uplink trigger, and
- reduces wireless latency and extends mobile device battery life

Additional information

- U-APSD activates automatically when WMM (Wi-Fi Multimedia) is enabled.
- U-APSD eliminates the need for polling individual packets at the access point.

Possible sequence for U-APSD operation:

A client device sends a single uplink trigger packet. In response, the access point delivers multiple queued downlink packets based on that single trigger.

View unscheduled automatic power save delivery

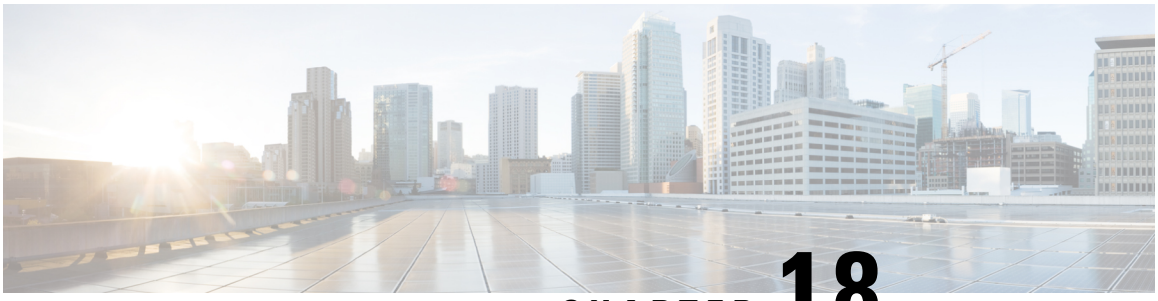
Show detailed information of a client by MAC address.

```
Device# show wireless client mac-address 2B:5B:B3:18:56:E9 detail
Output Policy State : Unknown
Output Policy Source : Unknown
WMM Support : Enabled
U-APSD Support : Enabled
U-APSD value : 15
APSD ACs    : BK(T/D), BE, VI(T/D), VO(T/D)
Power Save : OFF
Current Rate :

--------------------------
```

```
BK : Background
BE : Best Effort
VI : Video
VO : Voice.

T: UAPSD Trigger Enabled
D: UAPSD Delivery Enabled
T/D : UAPSD Trigger and Delivery Enabled
```



CHAPTER 18

Enabling USB Port on Access Points

- USB port as power source for access points, on page 149

USB port as power source for access points

A USB port as power source for access points is a power feature that

- provides up to 2.5W of power to connected USB devices
- automatically shuts down when a device draws more than 2.5W to prevent overload, and
- is available on select Cisco AP models as specified in their datasheets.

USB port power specifications

The USB port enables when the power draw is 2.5W and lower. Refer to the datasheet of your AP to check if the AP has a USB port that can act as a source of power.

Note The controller records the last five power-overdrawn incidents in its logs.

Caution When unsupported USB device is connected to the Cisco AP, the following message is displayed:

```
The inserted USB module is not a supported device. The behavior of this
USB device and the impact to the Access Point is not guaranteed. If Cisco
determines that a fault or defect can be isolated due to the use of
third-party USB modules installed by a customer or reseller, Cisco may
withhold support under warranty or support program under contract. In the
course of providing support for Cisco networking products, the end user
may be required to install Cisco-supported USB modules in the event Cisco
determines that removing third-party parts will assist Cisco in diagnosing
root cause for troubleshooting purposes. Cisco also reserves the right
to charge the customer per then-current time and material rates for
services provided to the customer when Cisco determines, after having
provided such services, that an unsupported device caused the root cause
of the defective product
```

Configure an AP profile (CLI)

Create and configure an AP profile to manage access point settings and behavior.

AP profiles allow you to define configuration parameters that can be applied to multiple access points, providing consistent settings and simplified management across your wireless network deployment.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter the AP profile configuration mode.

Example:

```
Device(config)# ap profile ap-profile
```

Example:

```
Device(config)# ap profile xyz-ap-profile
```

Note
When you delete a named profile, the APs associated with that profile will not revert to the default profile.

Step 3 Enable USB for each AP profile.

Example:

```
Device(config-ap-profile)# usb-enable
```

Note
By default, the USB for each AP profile is enabled.

Use the **no usb-enable** command to disable USB for each AP profile.

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config-ap-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The AP profile is now configured and can be applied to access points to manage their USB settings and other profile-specific configurations.

Configure USB settings for an access point (CLI)

Enable and manage USB functionality on access points to support connected USB devices or modules.

USB ports on access points can be used for various purposes including storage, cellular modules, or other peripheral devices. This configuration allows administrators to control USB port functionality at the individual AP level.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enable the USB port on the AP.

Example:

```
Device# ap name ap-name usb-module
```

Example:

```
Device# ap name AP44d3.xy45.69a1 usb-module
```

Use the **ap name** *ap-name* **no usb-module** command to disable the USB port on the AP.

Step 3 Override USB status of the AP profile and consider the local AP configuration.

Example:

```
Device# ap name ap-name usb-module override
```

Example:

```
Device# ap name AP44d3.xy45.69a1 usb-module override
```

Use the **ap name** *ap-name* **no usb-module override** command to override USB status of the AP and consider the AP profile configuration.

Note
You can configure the USB status for an AP only if you enable USB override for it.

The USB settings are now configured for the specified access point, allowing control over USB port functionality and override behavior.

USB configuration monitoring for access points (CLI)

Monitor USB configurations for access points using command-line interface commands to view inventory details, module summaries, configuration details, and status information.

- To view the inventory details of APs, use this command:

  show ap name *ap-name* **inventory**

  The following is a sample output:

```
Device# show ap name AP500F.8059.1620  inventory
NAME: AP2800    , DESCR: Cisco Aironet 2800 Series (IEEE 802.11ac) Access Point
PID: AIR-AP2802I-D-K9  ,  VID: 01,  SN: XXX1111Y2ZZZZ2800
NAME: SanDisk    , DESCR: Cruzer Blade
PID: SanDisk  ,  SN: XXXX1110010, MaxPower: 224
```

- To view the summary of an AP module, use this command:

  show ap module summary

  This is a sample output:

```
Device# show ap module summary
AP Name               External Module       External Module PID   External Module
Description
-------------------------------------------------------------------------------------------
AP500F.1111.2222      Enable                SanDisk               Cruzer Blade
```

- To view the USB configuration details for each AP, use this command:

  show ap name *ap-name* **config general**

  This is a sample output:

```
Device# show ap name AP500F.111.2222 config general

.
.
.
USB Module Type.................................. USB Module
USB Module Status................................ Disabled
USB Module Operational State..................... Enabled
USB Override ……..... Enabled
```

- To view status of the USB module, use this command:

  show ap profile name *xyz* **detailed**

  This is a sample output:

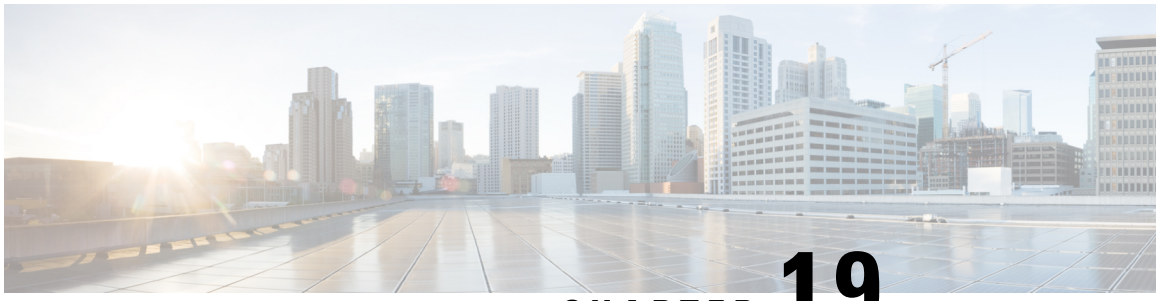
```
Device# show ap profile name xyz detailed
USB Module               : ENABLED
```



PART IV

Network Management

- DHCP Option82, on page 155
- RADIUS Realm, on page 169
- Persistent SSID Broadcast, on page 177
- Network Monitoring, on page 179



CHAPTER 19

DHCP Option82

- DHCP Option 82, on page 155
- Configure DHCP Option 82 global interface, on page 158
- Configure DHCP Option 82 through a VLAN interface, on page 160
- AP DHCP Option 82 support on FlexConnect local switching mode, on page 165

DHCP Option 82

DHCP Option 82 is a relay agent information option that

- carries information known by the relay agent
- increases DHCP security by enabling identification of request sources, and
- allows address allocation based on network topology or client device characteristics.

The controller can be configured to add Option 82 information to DHCP requests from clients before forwarding the requests to a DHCP server. The DHCP server can then be configured to allocate IP addresses to the wireless client based on the information present in DHCP Option 82.

DHCP provides a framework for passing configuration information to hosts on a TCP/IP network. Configuration parameters and other control information are carried in tagged data items stored in the Options field of the DHCP message. These data items are also called options. Option 82 includes information known by the relay agent.

The Relay Agent Information option is a single DHCP option containing one or more suboptions that convey information known by the Relay Agent. Option 82 enables a DHCP Relay Agent to insert circuit-specific information into a forwarded request to a DHCP server. This option uses two suboptions:

- Circuit ID: The Circuit ID suboption includes information that is specific to the circuit the request came in on. This suboption is an identifier that is specific to the relay agent. Thus, the circuit that is described will vary depending on the relay agent.
- Remote ID: The Remote ID suboption includes information on the remote host–end of the circuit. This suboption usually contains information that identifies the relay agent. In a wireless network, this is likely a unique identifier for the wireless AP.

Note

- IP MAC binding is required for DHCP Option 82 to work in some situations.

DHCP Option 82 configuration options

You can configure these DHCP Option 82 options in a controller:

- DHCP Enable
- DHCP Opt82 Enable
- DHCP Opt82 Ascii
- DHCP Opt82 RID
- DHCP Opt Format
- DHCP AP MAC
- DHCP SSID
- DHCP AP ETH MAC
- DHCP AP NAME
- DHCP Site Tag
- DHCP AP Location
- DHCP VLAN ID

The controller includes the SSID in ASCII format and the VLAN ID in hexadecimal format within the remote ID suboption of Option 82 in the outgoing DHCP packets to the server for this configuration.

```
ipv4 dhcp opt82 format ssid
ipv4 dhcp opt82 format vlan-id
```

However, if *ipv4 dhcp opt82 ascii* configuration is also present, the controller adds VLAN-ID and SSID in ASCII format.

For Cisco Catalyst 9800 Series Configuration Best Practices, see the following link: https://www.cisco.com/c/en/us/products/collateral/wireless/catalyst-9800-series-wireless-controllers/guide-c07-743627.html

Configure DHCP Option 82 format

Configure the wireless controller to insert DHCP Option 82 information into DHCP requests from wireless clients.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable configuration for the specified profile policy.

Example:

```
Device(config)# wireless profile policy policy-name
```

Step 3 Shut down the profile policy.

Example:

```
Device(config-wireless-policy)# shutdown
```

Step 4 Assign the profile policy to a VLAN.

Example:

```
Device(config-wireless-policy)# vlan vlan-name
```

Step 5 (Optional) Set the session timeout value in seconds. The range is between 20 to 86,400 seconds.

Example:

```
Device(config-wireless-policy)# session-timeout value-btwn-20-86400
```

Step 6 (Optional) Set the idle timeout value in seconds. The range is between 15 to 100,000 seconds.

Example:

```
Device(config-wireless-policy)# idle-timeout 15
```

Step 7 Enable central switching.

Example:

```
Device(config-wireless-policy)# central switching
```

Step 8 Enable DHCP Option 82 for the wireless clients.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82
```

Step 9 (Optional) Enable ASCII on the DHCP Option 82 feature.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 ascii
```

Step 10 (Optional) Add Cisco 2-byte Remote ID (RID) to the DHCP Option 82 feature.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 rid
```

Step 11 Enable DHCP Option 82 on the corresponding AP.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format {ap_ethmac | ap_location | apmac |
apname | policy_tag | ssid | vlan_id}
```

For information on the various options available with the command, see Cisco Catalyst 9800 Series Wireless Controller Command Reference .

Step 12 Enable the profile policy.

Example:

```
Device(config-wireless-policy)# no shutdown
```

The wireless profile policy inserts DHCP Option 82 information in the specified format for wireless clients.

```
Device# configure terminal
Device(config)# wireless profile policy pp3
Device(config-wireless-policy)# shutdown
Device(config-wireless-policy)# vlan 72
Device(config-wireless-policy)# session-timeout 300
Device(config-wireless-policy)# idle-timeout 15
Device(config-wireless-policy)# central switching
Device(config-wireless-policy)# ipv4 dhcp opt82
Device(config-wireless-policy)# ipv4 dhcp opt82 ascii
Device(config-wireless-policy)# ipv4 dhcp opt82 rid
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac
Device(config-wireless-policy)# no shutdown
```

Configure DHCP Option 82 global interface

Configure DHCP Option 82 globally through server override (CLI)

Set up DHCP Option 82 settings to override relay agent information in DHCP packets at the server.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Insert the global server override and link selection suboptions.

Example:

```
Device(config)# ip dhcp-relay information option server-override
```

The system processes DHCP Option 82 information globally based on server settings.

```
Device# configure terminal
Device(config)# ip dhcp-relay information option server-override
```

Configure DHCP Option 82 globally through different SVIs (GUI)

Enable DHCP Option 82 to insert relay information for clients across selected SVIs.

Procedure

Step 1 Choose **Configuration** > **VLAN**.

Step 2 Select a VLAN from the drop-down list.

The **Edit SVI** window appears.

Step 3 Click the **Advanced** tab.

Step 4 Select an option from the **IPv4 Inbound ACL** drop-down list.

Step 5 Select an option from the **IPv4 Outbound ACL** drop-down list.

Step 6 Select an option from the **IPv6 Inbound ACL** drop-down list.

Step 7 Select an option from the **IPv6 Outbound ACL** drop-down list.

Step 8 Enter an IP address in the **IPv4 Helper Address** field.

Step 9 Set the status to **Enabled** if you want to enable the **Relay Information Option** setting.

Step 10 Enter the **Subscriber ID**.

Step 11 Set the status to **Enabled** if you want to enable the **Server ID Override** setting.

Step 12 Set the status to **Enabled** if you want to enable the **Option Insert** setting.

Step 13 Select an option from the **Source-Interface Vlan** drop-down list.

Step 14 Click **Update & Apply to Device**.

DHCP Option 82 is enabled for the selected SVIs. This allows relay agent information to be included in DHCP requests.

Configure DHCP Option 82 globally through different SVIs (CLI)

Enable DHCP Option 82 globally so that DHCP relayed messages include option 82 information through the specified SVIs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Set global source interface for relayed messages.

Example:

```
Device(config)# ip dhcp-relay source-interface vlan vlan-id
```

The device globally applies DHCP relay with Option 82 information using the specified SVI interface.

```
Device# configure terminal
Device(config)# ip dhcp-relay source-interface vlan 72
```

Configure DHCP Option 82 through a VLAN interface

Configure DHCP Option 82 through option-insert command (CLI)

Enable DHCP Option 82 insertion on a VLAN interface to provide relay agent information in DHCP packets.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a VLAN ID.

Example:

```
Device(config)# interface vlan vlan-id
```

Step 3 Insert relay information in BOOTREQUEST.

Example:

```
Device(config-if)# ip dhcp relay information option-insert
```

Step 4 Configure the IP address for the interface.

Example:

```
Device(config-if)# ip address ip-address
```

Step 5 Configure the destination address for UDP broadcasts.

Example:

```
Device(config-if)# ip helper-address ip-address
```

Step 6 Disable the MOP for an interface.

Example:

```
Device(config-if)# no mop enabled
```

Step 7 Disable the task of sending MOP periodic system ID messages.

Example:

```
Device(config-apgroup)# no mop sysid
```

DHCP requests sent from this VLAN interface will contain Option 82 relay information to aid in DHCP server assignment and troubleshooting.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option-insert
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through server-ID-override command (CLI)

Enable DHCP Option 82 with server-ID override to control how DHCP relay information is inserted on a Cisco device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the server-id override suboption to use an RFC-specific or Cisco-specific value.

Example:

```
Device(config)# ip dhcp compatibility suboption server-override cisco
```

Step 3 Configure the link-selection suboption to use an RFC-specific or Cisco-specific value.

Example:

```
Device(config)# ip dhcp compatibility suboption link-selection cisco
```

Step 4 Configure a VLAN ID.

Example:

```
Device(config)# interface vlan vlan-id
```

Step 5 Insert the server-id override suboption and link-selection suboption.

Example:

```
Device(config-if)# ip dhcp relay information option server-id-override
```

Step 6 Configure the IP address for the interface.

Example:

```
Device(config-if)# ip address ip-address
```

Step 7 Configure the destination address for UDP broadcasts that your interfce will use.

Example:

```
Device(config-if)# ip helper-address ip-address
```

Step 8 Disable MOP for an interface.

Example:

```
Device(config-if)# no mop enabled
```

Step 9 Disable the task of sending MOP periodic system ID messages.

Example:

```
Device(config-if)# [no] mop sysid
```

The device uses the specified server-id override and link-selection options in DHCP Option 82 relay messages, which the device forwards through the configured VLAN interface.

```
Device# configure terminal
Device(config)# ip dhcp compatibility suboption server-override cisco
Device(config)# ip dhcp compatibility suboption link-selection cisco
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option server-id-override
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-if)# no mop sysid
```

Configure DHCP Option 82 through a subscriber-ID (CLI)

Configure DHCP Option 82 through a subscriber-ID.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a VLAN ID.

Example:

```
Device(config)# interface vlan vlan-id
```

Step 3 Insert the subscriber identifier suboption to include the subscriber ID in DHCP messages.

Example:

```
Device(config-if)# ip dhcp relay information option subscriber-id subscriber-id
```

Step 4 Configure the IP address for the interface.

Example:

```
Device(config-if)# ip address ip-address
```

Step 5 Configure the destination address for UDP broadcasts.

Example:

```
Device(config-if)# ip helper-address ip-address
```

Step 6 Disable MOP for an interface.

Example:

```
Device(config-if)# no mop enabled
```

Step 7 Disable the task of sending MOP periodic system ID messages.

Example:

```
Device(config-apgroup)# no mop sysid
```

The system now inserts the subscriber-ID suboption (DHCP Option 82) into relayed DHCP packets on the specified VLAN interface.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option subscriber-id test10
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 92.0.2.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through server-ID-override and subscriber-ID commands (CLI)

Enable customized handling of DHCP Option 82 information for relayed DHCP requests by setting server and subscriber ID options.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a VLAN ID.

Example:

```
Device(config)# interface vlan vlan-id
```

Step 3 Insert server ID override and link selection suboptions.

Example:

```
Device(config-if)# ip dhcp relay information option server-id-override
```

Step 4 Insert the subscriber identifier suboption.

Example:

```
Device(config-if)# ip dhcp relay information option subscriber-id subscriber-id
```

Step 5 Configure the IP address for the interface.

Example:

```
Device(config-if)# ip address ip-address
```

Step 6 Configure the destination address for UDP broadcasts.

Example:

```
Device(config-if)# ip helper-address ip-address
```

Step 7 Disable the MOP for an interface.

Example:

```
Device(config-if)# no mop enabled
```

Step 8 Disable the task of sending MOP periodic system ID messages.

Example:

```
Device(config-apgroup)# no mop sysid
```

The interface relays DHCP requests with Option 82 data, including server ID override and subscriber ID.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option server-id-override
Device(config-if)# ip dhcp relay information option subscriber-id test10
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through different SVIs (CLI)

Enable DHCP Option 82 using different switch virtual interfaces (SVIs) for relayed requests.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a VLAN ID.

Example:

```
Device(config)# interface vlan vlan-id
```

Step 3 Configure a source interface on the VLAN to relay messages.

Example:

```
Device(config-if)# ip dhcp relay source-interface vlan vlan-id
```

Step 4 Configure the IP address for the interface.

Example:

```
Device(config-if)# ip address ip-address
```

Step 5 Configure the destination address for UDP broadcasts.

Example:

```
Device(config-if)# ip helper-address ip-address
```

Step 6 Disable the Maintenance Operations Protocol (MOP) on the interface.

Example:

```
Device(config-if)# no mop enabled
```

Step 7 Disable periodic system ID message transmission for MOP.

Example:

```
Device(config-apgroup)# no mop sysid
```

The device relays DHCP messages with Option 82 through the configured VLAN SVIs.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay source-interface vlan 74
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

AP DHCP Option 82 support on FlexConnect local switching mode

AP DHCP Option 82 support on FlexConnect local switching mode is a DHCP relay feature that

- provides additional information on the physical attachment of the client
- enhances security while using DHCP for network address allocation, and
- enables the AP to act as a DHCP relay agent to prevent DHCP client requests from unreliable sources.

DHCP Option 82 functionality

As a DHCP relay agent, the AP can add DHCP Option 82 information such as AP MAC, AP Name, and SSID to DHCP requests from clients before forwarding the requests to the DHCP server. Afterward, the DHCP servers can allocate IP addresses to wireless clients based on the data contained within DHCP Option 82.

Configure AP DHCP Option82 support

Enable DHCP Option82 support on access points to provide additional network information to DHCP servers.

DHCP Option82 allows access points to insert additional information such as AP MAC address, AP name, or SSID into DHCP requests from clients. This information helps DHCP servers make more informed decisions about IP address assignments and network policies.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable configuration for the specified profile policy.

Example:

```
Device(config)# wireless profile policy policy-name
```

Example:

```
Device(config)# wireless profile policy policy-ap
```

Step 3 Shut down the profile policy.

Example:

```
Device(config-wireless-policy)# shutdown
```

Step 4 Configure the AP MAC address or the AP name format to be added to DHCP requests from clients before forwarding them to the DHCP server.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac|apname
```

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac
```

You can use either apmac or apname, but not both.

Step 5 Configure the unique SSID format to be added to DHCP requests from clients before forwarding them to the DHCP server.

Example:

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format ssid
```

Step 6 Enable the profile policy.

Example:

```
Device(config-wireless-policy)# no shutdown
```

DHCP Option82 support is now configured on the access point policy, allowing the AP to insert additional network information into DHCP requests from connected clients.

Verify AP DHCP Option82 support

Verify the configured parameters for DHCP Option82 using commands on the AP console and controller to confirm the configuration status and settings.

To verify the configured parameters for DHCP Option82, use this command on the AP console:

Note You can fetch information about the WLAN and WLAN ID using the **show flexconnect WLAN** command.

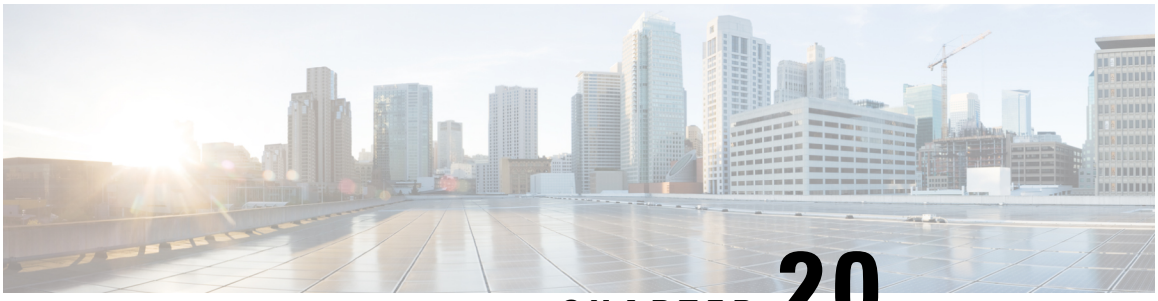
```
Device# show flexconnect dhcp option-82 wlan 1

DHCP OPTION 82 CONFIG FOR WLAN:
VAP ID      :0
Format      :Binary
Delimiter   :":"
Circuit-ID  :
Remote-ID   :
    AP Name
    SSID-Name
```

To verify whether DHCP Option82 is enabled on the controller, use this command:

```
Device# show wireless profile policy detailed

Opt82
  SSID                    : ENABLED
  AP_ETHMAC               : DISABLED
  APNAME                  : ENABLED
```



CHAPTER 20

RADIUS Realm

- RADIUS realms, on page 169

RADIUS realms

A RADIUS realm is an AAA configuration element that

- enables assigning authentication and accounting requests to specific RADIUS servers based on user domain information
- uses the domain portion of a user's Network Access Identifier (NAI) to select the appropriate RADIUS server, and
- provides realm-based filtering and control for authentication and accounting requests within a WLAN.

Feature history

Table 11: Feature history for RADIUS realms

Feature name	Release information	Feature description
RADIUS realms	Cisco IOS XE 16.9.1	RADIUS realms are configuration elements in AAA (Authentication, Authorization, and Accounting) systems. They use the domain portion of a user's Network Access Identifier (NAI)—such as the part after "@" in an email address—to direct authentication and accounting requests to specific RADIUS servers. Realms help organizations manage user access and resource usage across different groups or domains.

When mobile clients connect to a WLAN, the RADIUS realm is included in the Extensible Authentication Protocol Method for UMTS Authentication and Key Agreement (EAP-AKA) identity response of the authentication request packet. For WLAN, the NAI format for EAP-AKA can be specified as *username@domain.com*. In the NAI format, the realm follows the @ symbol and is specified as *example.com*. If the vendor-specific attribute "test" is added, the NAI format becomes *test@example.com*.

The RADIUS Realm feature can be enabled and disabled on a WLAN. If Realm is enabled on a WLAN, the corresponding user should send the username in the NAI format. The controller sends the authentication request to the AAA server only after it receives a realm in NAI format from the client and verifies that the

realm complies with the required standards. Additionally, the controller sends accounting requests to the AAA server based on realm filtering.

Realm support on a WLAN

Each WLAN is configured to support NAI realms. Once the realm is enabled on a specific SSID, the system compares the received realms in the EAP identity response to those configured on the RADIUS server. If a username with the realm is not sent by the client, the WLAN uses the default RADIUS server for authentication. When the client's realm does not match the realms configured on the WLAN, the client is deauthenticated and dropped.

If the RADIUS Realm feature is not enabled on a WLAN, the controller uses the username received in the EAP identity request directly and authenticates the user using the configured RADIUS server. The RADIUS Realm feature is disabled on WLANs by default.

- **Realm match for authentication**: In dot1x with EAP methods (similar to EAP AKA), the username is received as part of an EAP identity response. A realm is derived from the username and is matched with the realms that are already configured in the corresponding RADIUS authentication server. If there is a match, the authentication requests are forwarded to the RADIUS server. If there is a mismatch, the client is deauthenticated.
- **Realm match for accounting**: After receiving a client's username through an access-accept message, the system derives the realm from the username when accounting messages are triggered. The realm is compared with the configured accounting realms on the RADIUS accounting server. When there is a match, the server receives the accounting requests. If there is no match, the system drops the accounting requests.

Enable RADIUS realm

Enable the RADIUS realm to allow the WLAN to use the RADIUS realm selection for AAA authentication.

Enabling the RADIUS realm is required when configuring wireless AAA policies that use RADIUS for authentication on your Cisco device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new AAA policy.**wireless aaa policy**

Example:

```
Device(config)# wireless aaa policy aaa-policy
```

Step 3 Enable AAA RADIUS realm selection.

Example:

```
Device(config-aaa-policy)# aaa-realm enable
```

Note

Use the **no aaa-realm enable** or the **default aaa-realm enable** command to disable the RADIUS realm.

The AAA RADIUS realm is now enabled and available for authentication purposes in the specified wireless AAA policy.

```
Device# configure terminal
Device(config)# wireless aaa policy policy-1
Device(config-aaa-policy)# aaa-realm enable
```

Configure realm to match the RADIUS server for authentication and accounting

Configure the realm to use your chosen RADIUS server group for authentication and accounting network access requests.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a AAA authentication model.

Example:

```
Device(config)# aaa new-model
```

Step 3 Set the authorization method.

Example:

```
Device(config)# aaa authorization network default group radius-server-group
```

Step 4 Indicate that 802.1x must use the realm group RADIUS server.

Example:

```
Device(config)# aaa authentication dot1x realm group radius-server-group
```

Step 5 Define the authentication method at login.

Example:

```
Device(config)# aaa authentication login realm group radius-server-group
```

Step 6 Enable accounting so the system sends a start record notice when a client is authorized and a stop record notice when the session ends.

Example:

```
Device(config)# aaa accounting identity realm start-stop group radius-server-group
```

Your device now uses the realm and RADIUS server group for authentication, authorization, and accounting.

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization network default group aaa_group_name
Device(config)# aaa authentication dot1x cisco.com group cisco1
Device(config)# aaa authentication login cisco.com group cisco1
Device(config)# aaa accounting identity cisco.com start-stop group v
```

Configure AAA policy for WLANs

Configure authentication, authorization, and accounting (AAA) policies to control and secure access for WLAN users.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new AAA policy for wireless.

Example:

```
Device(config)# wireless aaa policy aaa-policy-name
```

Step 3 Enable AAA RADIUS server selection by realm.

Example:

```
Device(config-aaa-policy)# aaa-realm enable
```

Step 4 Return to global configuration mode.

Example:

```
Device(config-aaa-policy)# exit
```

Step 5 Configure a WLAN policy profile.

Example:

```
Device(config)# wireless profile policy wlan-policy-profile
```

Step 6 Map the AAA policy.**aaa-policy***aaa-policy*

Example:

```
Device(config-wireless-policy)# aaa-policy aaa-policy
```

Step 7 Set the accounting list.**accounting-list***acct-config-realm*

Example:

```
Device(config-wireless-policy)# accounting-list acct-config-realm
```

Step 8 Return to global configuration mode.

Example:

```
Device(config-wireless-policy)# exit
```

Step 9 Configure a WLAN.**wlan***wlan-namewlan-idssid*

Example:

```
Device(config)# wlan wlan-name wlan-id ssid
```

Step 10 Enable the security authentication list for IEEE 802.1x.

Example:

```
Device(config-wlan)# security dot1x authentication-list auth-list-realm
```

Step 11 Return to global configuration mode.

Example:

```
Device(config-wlan)# exit
```

Step 12 Configuresa policy tag.

Example:

```
Device(config)# wireless tag policy policy-name
```

Step 13 Map a policy profile to the WLAN.

Example:

```
Device(config-policy-tag)# wlan wlan-name policy policy-profile
```

Step 14 Return to global configuration mode.

Example:

```
Device(config-policy-tag)# exit
```

The AAA policy is successfully applied to the target WLAN, enabling centralized authentication and accounting for users connecting to that WLAN.

```
Device# configure terminal
Device(config)# wireless aaa policy aaa-policy-1
Device(config-aaa-policy)# aaa-realm enable
Device(config-aaa-policy)# exit
Device(config)# wireless profile policy wlan-policy-1
Device(config-wireless-policy)# aaa-policy aaa-policy-1
Device(config-wireless-policy)# accounting-list cisco.com
Device(config-wireless-policy)# exit
Device(config)# wlan wlan2 14 wlan-aaa
Device(config-wlan)# security dot1x authentication-list cisco.com
Device(config-wlan)# exit
Device(config)# wireless tag policy tag-policy-1
Device(config-policy-tag)# wlan Abc-wlan policy wlan-policy-a
Device(config-policy-tag)# exit
```

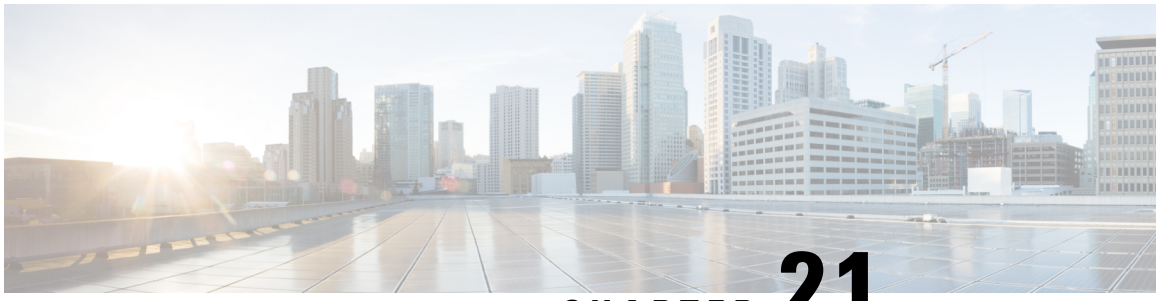
Verify RADIUS-realm configuration

Run this command to verify the RADIUS-realm configuration.

```
Device# show wireless client mac-address 14bd.61f3.6a24 detail
Client MAC Address : 14bd.61f3.6a24
Client IPv4 Address : 192.0.2.1
Client IPv6 Addresses : fe80::286e:9fe0:7fa6:8f4
```

```
Client Username : cisco-mac@cisco.com
AP MAC Address : 4c77.6d79.5a00
AP Name: AP4c77.6d53.20ec
AP slot : 1
Client State : Associated
Policy Profile : name-policy-profile
Flex Profile : N/A
Wireless LAN Id : 3
Wireless LAN Name: ha_realm_WLAN_WPA2_AES_DOT1X
BSSID : 4c77.6d79.5a0f
Connected For : 26 seconds
Protocol : 802.11ac
Channel : 44
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Re-Authentication Timeout : 1800 sec (Remaining time: 1775 sec)
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 06/12/2018 19:52:35 IST
Policy Manager State: Run
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 25 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
Encrypted Traffic Analytics : No
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : PEAP
VLAN : 113
Multicast VLAN : 0
Access VLAN : 113
Anchor VLAN : 0
WFD capable : No
Managed WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Interface        : capwap_9040000f
  IIF ID           : 0x9040000F
  Authorized       : TRUE
  Session timeout  : 1800
  Common Session ID: 097704090000000DF4607B3B
  Acct Session ID  : 0x00000fa2
  Aaa Server Details
  Server IP        : 192.0.2.2
```

```
  Auth Method Status List
       Method : Dot1x
              SM State         : AUTHENTICATED
              SM Bend State    : IDLE
  Local Policies:
       Service Template : wlan_svc_name-policy-profile_local (priority 254)
              Absolute-Timer   : 1800
              VLAN             : 113
  Server Policies:
  Resultant Policies:
              VLAN             : 113
              Absolute-Timer   : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Central
FlexConnect Dhcp Status : Central
FlexConnect Authentication : Central
FlexConnect Central Association : No
.
.
.
Fabric status : Disabled
Client Scan Reports
Assisted Roaming Neighbor List
```



CHAPTER 21

Persistent SSID Broadcast

This chapter explains the persistent SSID broadcast feature for Cisco Aironet 1542 RAPs and MAPs, providing command steps to enable or disable it and commands to verify AP configuration and behavior when the WAN link is down.

- Persistent SSID broadcast, on page 177

Persistent SSID broadcast

A persistent SSID broadcast is a wireless network feature that

- enables Root APs (RAPs) and Mesh APs (MAPs) to continue broadcasting the SSID when the WAN link is down
- helps operators isolate issues between the backhaul network and the access wireless network, and
- applies specifically to Cisco Aironet 1542 APs operating in FlexConnect plus Bridge mode.

RAPs have a wired connection to the embedded wireless controller while MAPs have wireless connection to the embedded wireless controller.

This requirement helps isolate responsibility, whether the fault lies with the backhaul or the access wireless network. Different operators may own each part of the network.

RAPs and MAPs broadcast the SSID while in standalone mode if the default gateway is reachable.

Also refer Mesh Deployment Guide for Cisco Catalyst 9800 Series Wireless Controllers.

Configure persistent SSID broadcast (CLI)

Enable or disable persistent SSID broadcast so that associated APs re-join and maintain SSID visibility as configured using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the AP profile.

Example:

```
Device(config)# ap profile ap-profile-name
```

Step 3 Configure the SSID broadcast mode.

Example:

```
Device(config-ap-profile)# ssid broadcast persistent

Device(config-ap-profile)# [no] ssid broadcast persistent
```

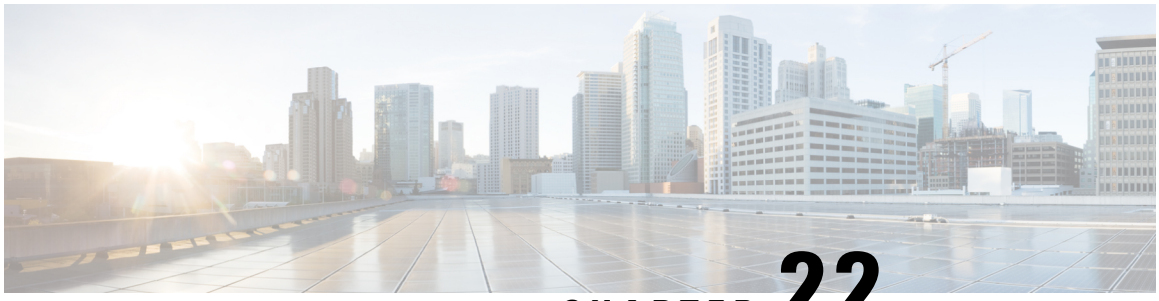
The **persistent** keyword enables a persistent SSID broadcast, where the associated APs re-join. Use the **no** form of the command to disable the feature.

Note
Enabling or disabling this feature causes the AP to re-join.

Verify persistent SSID broadcast

To view the configuration of all Cisco APs, use this **show** command:

```
Device#show ap config general
Cisco AP Name   : AP4C77.6DF2.D598
================================================
Office Extend Mode                              : Disabled
Persistent SSID Broadcast                       : Enabled
Remote AP Debug                                 : Disabled
```



CHAPTER 22

Network Monitoring

- Network monitoring , on page 179

Network monitoring

A network monitoring system is a network management solution that

- collects and exposes real-time operational data from network devices through APIs
- uses protocols such as NETCONF/YANG, and other technologies to automate data transfer and enable programmability, and
- provides synchronous status updates and asynchronous notifications for alarms, events, and third-party applications.

Network monitoring systems can expose APIs or push data to third-party applications. In Cisco environments, data transfer commonly occurs through the NETCONF/YANG protocol, supporting automated and programmable network operations.

- The two types of information provided are:
  - Status information received synchronously: NETCONF is the management interface for status information received synchronously. It allows publishing the operational state of the device, including the controller.
  - Alarm and event information sent asynchronously: NETCONF/YANG push is used for alarm and event information. It provides the mechanism for sending subscribed NETCONF notifications.
- Contact API or Developer Support for NETCONF/YANG features at this link:

  https://developer.cisco.com/site/support/#

Note When using NETCONF, you may see a user named \"yang_mgmt_infra"\ appearing in accounting records for TACACS or RADIUS. This user is internal. If you pull data from Cisco IOS, you should expect to see this user.

Status information received synchronously - configuration examples

You can use the NETCONF/YANG interface to fulfill customer requests.

To enable Status Information, Alarm Information, and Event Information, configure the NETCONF server on the controller using this command.

netconf-yang

Note The Cisco Catalyst 9800 wireless controller supports only RSA keys for the trustpoint used by the ncsshd process. EC keys are not supported; using them causes the ncsshd process to crash and prevents you from using NETCONF.

The **netconf-yang** command enables notifications, configuration access, and operation access (OAM) through NETCONF/YANG. For details about the NETCONF/YANG, refer to the *NETCONF Protocol* chapter of the Programmability Configuration Guide at https://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-xe-17/products-installation-and-configuration-guides-list.html

For the Status Information Received Synchronously type, NETCONF exports these items:

- Name of the village
- APs in each village
- Status of each AP
- Number of clients currently connected and logged on in each village and each AP

All the data for the items listed above is already available as the controller operational data exported through NETCONF. The examples below explain where the data items listed are available.

This command is used in the controller:

```
wireless tag site village_name_1
```

The site tags can be retrieved by NETCONF using the **get-config** operation.

Example output for **Name of the Village**:

```
<site-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-site-cfg">
[...]
<site-tag-configs>
  <site-tag-config>
    <site-tag-name>village_name_1</site-tag-name>
    <description>custom user site tag for a village</description>
  </site-tag-config>
[...]
</site-tag-configs>
```

The operational data for the controller lists all connected (joined) APs and their corresponding site tags. Each output entry includes detailed information about the access point and its site tag. The relevant fields and controller show commands are presented here:

Example output of **AP per Village**:

```
<data>
    <access-point-oper-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-oper">
    [...]
```

```
      <radio-oper-data>
        <wtp-mac>00:1b:0c:00:02:00</wtp-mac>   #show ap dot11 {24ghz|5ghz} summary "MAC
Address"
        <radio-slot-id>0</radio-slot-id>       #show ap dot11 {24ghz|5ghz} summary "Slot"
        <ap-mac>00:1b:0c:00:02:00</ap-mac>
        <slot-id>0</slot-id>
        <radio-type>1</radio-type>             # 1 - 2.4GHz, 2 - 5GHz
        <admin-state>enabled</admin-state>     #show ap dot11 {24ghz|5ghz} summary "Admin
State"
        <oper-state>radio-up</oper-state>      #show ap dot11 {24ghz|5ghz} summary "Oper
State"
    [...]
[...]
      <capwap-data>
        <wtp-mac>00:1b:0c:00:02:00</wtp-mac>                #show ap summary  "Radio MAC"
        <ap-operation-state>registered</ap-operation-state> #show ap summary  "State"
        <ip-addr>10.102.140.10</ip-addr>                    #show ap summary  "IP Address"
        [...]
        <admin-state>1</admin-state>              #show ap status   "Status", 1 - Enabled,
 2 - Disabled
        <location>default-location </location>    #show ap summary  "Location"
        <country-code>CH </country-code>
        <name>AP_A-1</name>                       #show ap summary  "AP Name"
 [...]
        <tag-info>
          [...]
          <site-tag>
           <site-tag-name>village_name_1</site-tag-name> #show ap name AP_A-1 config general
 "Site Tag Name"
            [...]
          </site-tag>
[...]
```

The operational data of the controller contains information about all connected wireless clients. This includes detailed device-related information, such as MAC address, IP address, state, and access point name.

Example output of the **Number of clients currently online and logged in each village and each AP**:

```
<data>
    <client-oper-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-client-oper">
      <common-oper-data>
        <client-mac>00:00:1a:04:00:02</client-mac>     #show wireless client summary  "MAC
 Address"
        <ap-name>AP_A-1</ap-name>                      #show wireless client summary  "AP
Name"
        [...]
       <co-state>client-status-run</co-state>        #show wireless client summary  "State"
```

Alarm and Event Information Received Asynchronously - Configuration Examples

The push functionality for the alarm and event information is fulfilled with on-change notifications through NETCONF dynamic subscriptions, with XML encoding.

Example output of **AP Up/Down Events - Subscription**

```
Request:
```

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="urn:uuid:b0c581c9-ff5a-4352-9e64-7f2ce1ec603a"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <establish-subscription xmlns="urn:ietf:params:xml:ns:yang:ietf-event-notifications"
   xmlns:yp="urn:ietf:params:xml:ns:yang:ietf-yang-push">
  <stream>yp:yang-push</stream>
  <yp:xpath-filter>/access-point-oper-data/capwap-data/ap-operation-state</yp:xpath-filter>

  <yp:dampening-period>0</yp:dampening-period>
 </establish-subscription>
</rpc>

Reply:

<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:673b42b2-e988-4e20-a6c3-0679c08e6114"><subscription-result
xmlns='urn:ietf:params:xml:ns:yang:ietf-event-notifications'
xmlns:notif-bis="urn:ietf:params:xml:ns:yang:ietf-event-notifications">notif-bis:ok</subscription-result>
<subscription-id
xmlns='urn:ietf:params:xml:ns:yang:ietf-event-notifications'>2147483652</subscription-id>
</rpc-reply>
-->>
(Default Callback)
Event time      : 2018-03-09 15:08:21.880000+00:00
Subscription Id : 2147483651
Type            : 2
Data            :
<datastore-changes-xml xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push">
  <yang-patch xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-patch">
    <patch-id>null</patch-id>
    <edit>
      <edit-id>edit1</edit-id>
      <operation>merge</operation>
      <target>/access-point-oper-data/capwap-data</target>
      <value>
       <capwap-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-oper">

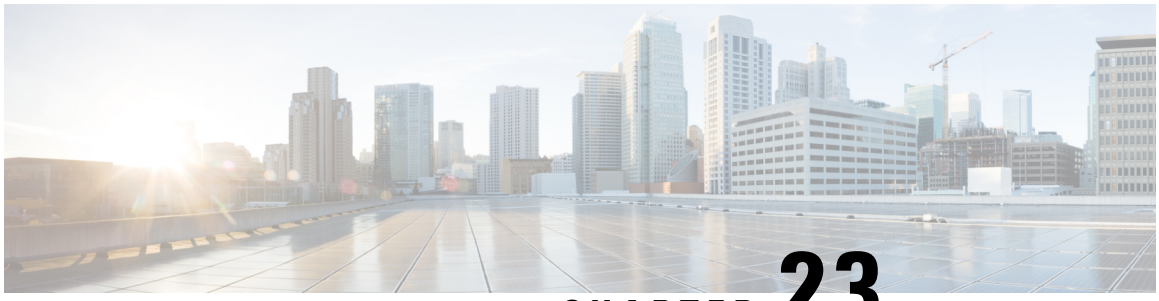
          <ap-operation-state>registered</ap-operation-state>
          <wtp-mac>00ab11006600</wtp-mac>
        </capwap-data>
      </value>
    </edit>
  </yang-patch>
</datastore-changes-xml>
<<--
```



PART V

System Management

- Network Mobility Services Protocol, on page 185
- Application Visibility and Control, on page 197
- Cisco Spaces , on page 219
- EDCA Parameters, on page 223
- 802.11 parameters and Band Selection, on page 225
- Image Download, on page 241
- Conditional Debug, Radioactive Tracing, and Packet Tracing, on page 253
- Aggressive Client Load Balancing, on page 265
- Accounting Identity List, on page 269
- Volume metering, on page 273
- Enabling Syslog Messages in Access Points and Controller for Syslog Server, on page 275



CHAPTER 23

Network Mobility Services Protocol

- Information About Network Mobility Services Protocol, on page 185
- Enabling NMSP On-Premises Services, on page 186
- Modifying the NMSP Notification Interval for Clients, RFID Tags, and Rogues , on page 186
- Modifying the NMSP Notification Threshold for Clients, and Tags, on page 187
- Configure NMSP strong cipher (CLI), on page 187
- Verify NMSP settings, on page 188
- Examples: NMSP settings configuration, on page 190
- Probe RSSI location, on page 190
- Configuring Probe RSSI , on page 191
- Verifying Probe RSSI, on page 192
- RFID tag support, on page 192
- Configure RFID tag support (CLI), on page 193
- Verify RFID tag support, on page 193

Information About Network Mobility Services Protocol

Cisco Network Mobility Services Protocol (NMSP) is a secure two-way protocol that can be run over a connection-oriented (TLS) or connection-less (DTLS) transport. The wireless infrastructure runs the NMSP server and Cisco Connected Mobile Experiences (Cisco CMX) acts as an NMSP client. The embedded wireless controller supports multiple services and multiple Cisco CMXs can connect to the NMSP server to get the data for the services (location of wireless devices, probe RSSI, hyperlocation, wIPS, and so on.) over the NMSP session.

NMSP defines the intercommunication between Cisco CMX and the embedded wireless controller. Cisco CMX communicates to the embedded wireless controller over a routed IP network. Both publish-subscribe and request-reply communication models are supported. Typically, Cisco CMX establishes a subscription to receive services data from the embedded wireless controller in the form of periodic updates. The embedded wireless controller acts as a data publisher, broadcasting services data to multiple CMXs. Besides subscription, Cisco CMX can also send requests to the embedded wireless controller, causing the embedded wireless controller to send a response back.

NMSP essentially provides a way to the applications in the embedded wireless controller to talk to the outside world. The NMSP in the embedded wireless controller also provides the flexibility to change the protocol to talk to the outside world.

The following is a list of the Network Mobility Services Protocol features:

- NMSP is disabled by default.
- NMSP communicates with Cisco CMX using TCP, and uses TLS for encryption.



Note HTTPS is not supported for data transport between embedded wireless controller and Cisco CMX.

Enabling NMSP On-Premises Services

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**nmsp enable** **Example:** `Device(config)# nmsp enable`	Enables NMSP on premises services. **Note** By default, the NMSP is disabled on the embedded wireless controller.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Modifying the NMSP Notification Interval for Clients, RFID Tags, and Rogues

NMSP manages communication between the Cisco Connected Mobile Experiences (Cisco CMX) and the embedded wireless controller for incoming and outgoing traffic. If your application requires more frequent location updates, you can modify the NMSP notification interval (to a value between 1 and 180 seconds) for clients, active RFID tags, and rogue access points and clients.



Note The TCP port (16113) that the embedded wireless controller and Cisco CMX communicate over must be open (not blocked) on any firewall that exists between the embedded wireless controller and the Cisco CMX for NMSP to function.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Modifying the NMSP Notification Threshold for Clients, and Tags

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**location notify-threshold** {**clients** | **tags** } *threshold* **Example:** `Device(config)# location notify-threshold clients 5`	Configures the NMSP notification threshold for clients, and tags. *threshold*- RSSI threshold value in db. Valid range is from 0 to 10.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure NMSP strong cipher (CLI)

Enable strong cipher suites for NMSP server to enhance security for device communication.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable strong ciphers for NMSP server.

Example:

```
Device(config)# nmsp strong-cipher
```

Strong ciphers for NMSP server contains, "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, AES256-SHA256:AES256-SHA:, and AES128-SHA256:AES128-SHA".

Normal cipher suite contains, "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, and AES128-SHA".

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, press ***Ctrl-Z*** to exit global configuration mode.

Verify NMSP settings

To view the NMSP capabilities of the embedded wireless controller, use the command:

```
Device# show nmsp capability
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum           Aggregate Interferer, Air Quality, Interferer,
Info               Rogue, Mobile Station,
Statistics         Rogue, Tags, Mobile Station,
AP Monitor         Subscription
On Demand Services Device Info
AP Info            Subscription
```

To view the NMSP notification intervals, use the command:

```
Device# show nmsp notification interval
NMSP Notification Intervals
---------------------------

RSSI Interval:
Client              : 2 sec
RFID                : 50 sec
Rogue AP            : 2 sec
Rogue Client        : 2 sec
Spectrum            : 2 sec
```

To view the connection-specific statistics counters for all CMX connections, use the command:

```
Device# show nmsp statistics connection
NMSP Connection Counters
------------------------
 CMX IP Address: 10.22.244.31, Status: Active
 State:
 Connections : 1
```

```
Disconnections : 0
Rx Data Frames : 13
Tx Data Frames : 99244
Unsupported messages : 0
Rx Message Counters:
ID  Name                                  Count
---------------------------------------------
1  Echo Request                          6076
7  Capability Notification                  2
13  Measurement Request                     5
16  Information Request                     3
20  Statistics Request                      2
30  Service Subscribe Request               1

Tx Message Counters:
ID  Name                                  Count
---------------------------------------------
2  Echo Response                          6076
7  Capability Notification                   1
14  Measurement Response                    13
15  Measurement Notification              91120
17  Information Response                      6
18  Information Notification               7492
21  Statistics Response                       2
22  Statistics Notification                 305
31  Service Subscribe Response                1
67  AP Info Notification                    304
```

To view the common statistic counter of the embedded wireless controller's NMSP service, use the command:

```
Device# show nmsp statistics summary
NMSP Global Counters
--------------------
Number of restarts                :

SSL Statistics
--------------------
Total amount of verifications      : 6
Verification failures              : 6
Verification success               : 0
Amount of connections created      : 8
Amount of connections closed       : 7
Total amount of accept attempts    : 8
Failures in accept                 : 0
Amount of successful accepts       : 8
Amount of failed registrations     : 0

AAA Statistics
--------------------
Total amount of AAA requests       : 7
Failed to send requests            : 0
Requests sent to AAA               : 7
Responses from AAA                 : 7
Responses from AAA to validate     : 7
Responses validate error           : 6
Responses validate success         : 1
```

To view the overall NMSP connections, use the command:

```
Device# show nmsp status
NMSP Status
-----------

CMX IP Address   Active    Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data     Transport
```

```
-------------------------------------------------------------------------------------------
127.0.0.1        Active    6             6             1           2           TLS
```

To view all mobility services subscribed by all CMXs, use the command:

```
Device# show nmsp subscription detail
CMX IP address 127.0.0.1:
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

To view all mobility services subscribed by a specific CMX, use the command:

```
Device# show nmsp subscription detail
CMX IP address 127.0.0.1:
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription

Device# show nmsp subscription summary
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

Examples: NMSP settings configuration

This example shows how to configure the NMSP notification interval for RFID tags:

```
Device# configure terminal
Device(config)# nmsp notification interval rssi rfid 50
Device(config)# end
Device# show nmsp notification interval
```

This example shows how to configure the NMSP notification interval for clients:

```
Device# configure terminal
Device(config)# nmsp notification interval rssi clients 180
Device(config)# end
Device# show nmsp notification interval
```

Probe RSSI location

A probe RSSI location is a wireless networking feature that

- allows wireless controllers and Cisco CMX to collect and use the received signal strength indication (RSSI) from client probe requests associated with the identified Service Set Identifiers (SSIDs)
- enables functions such as load balancing, coverage hole detection, and location updates, and
- supports the precise tracking of wireless clients as they move within the network.

After connecting to an AP, the client scans in the background and keeps a list of strong APs. If it loses connection, the client uses this list to reconnect. The APs in the WLAN gather probe requests, RSSI data, and MAC addresses of wireless clients, then forward this information to the wireless embedded wireless controllers. Cisco CMX receives this data from the embedded wireless controller and calculates the client's updated location.

Configuring Probe RSSI

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless probe filter** **Example:** `Device(config)# wireless probe filter`	Enables filtering of unacknowledged probe requests from AP to improve the location accuracy. Use the **no** form of the command to disable the feature. This will forward both acknowledged and unacknowledged probe requests to the embedded wireless controller.
Step 3	**wireless probe limit** *limit-value interval* **Example:** `Device(config)# wireless probe limit 10 100`	Configures the number of probe request reported to the wireless embedded wireless controller from the AP for the same client on a given interval. Use the **no** form of the command to revert to the default limit, which is 2 probes at an interval of 500 ms.
Step 4	**wireless probe locally-administered-mac** **Example:** `Device(config)# wireless probe locally-administered-mac`	Enables the reporting of probes from clients having locally administered MAC address.
Step 5	**location algorithm rssi-average** **Example:** `Device(config)# location algorithm rssi-average`	Sets the probe RSSI measurement updates to a more accurate algorithm but with more CPU overhead.

	Command or Action	Purpose
Step 6	**location algorithm simple** **Example:** `Device(config)# location algorithm simple`	(Optional) Sets the probe RSSI measurement updates to a faster algorithm with smaller CPU overhead, but less accuracy. Use the **no** form of the command to revert the algorithm type to the default one, which is *rssi-average*.
Step 7	**location expiry client** *interval* **Example:** `Device(config)# location expiry client 300`	Configures the timeout for RSSI values. The **no** form of the command sets it to a default value of 15.
Step 8	**location notify-threshold client** *threshold-db* **Example:** `Device(config)# location notify-threshold client 5`	Configures the notification threshold for clients. The **no** form of the command sets it to a default value of 0.
Step 9	**location rssi-half-life client** *time-in-seconds* **Example:** `Device(config)# location rssi-half-life client 20`	Configures half life when averaging two RSSI readings. To disable this option, set the value to 0.

What to do next

Use the **show wireless client probing** command to view each probing client (associated and probing only) by batch of 10 mac addresses.

Verifying Probe RSSI

To view the details of the AP the associated client was detected with, and with which RSSI:

```
Device# show wireless client mac-address 4.4.4 detail
****snippet of the output****
Nearby AP Statistics:
TEST_AP-1 (slot 0)
antenna 0: 0 s ago ........ -77 dBm
antenna 1: 0 s ago ........ -88 dBm
TEST_AP-5 (slot 0)
antenna 0: 0 s ago ........ -64 dBm
antenna 1: 0 s ago ........ -36 dBm
TEST_AP-6 (slot 0)
antenna 0: 0 s ago ........ -69 dBm
antenna 1: 0 s ago ........ -79 dBm
```

RFID tag support

An RFID tag is a small wireless battery-powered tracking device that

- broadcasts its own signal for real-time location tracking, and
- operates using special 802.11 packets processed by APs, wireless controllers, and Cisco CMX.

The embedded wireless controller enables you to configure radio frequency identification (RFID) tag tracking. Only active RFIDs are supported. A combination of active RFID tags and wireless embedded wireless controller allows you to track the current location of equipment. Use *Active* tags to track high-value assets in *closed-loop* systems. These tags remain within the control premises of the owner.

General guidelines

- You can verify the RFID tags on the embedded wireless controller.
- The controller supports high availability for RFID tags.

Configure RFID tag support (CLI)

Enable and configure RFID tag tracking features for WLANs managed by the device using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable RFID tag tracking.

Example:

```
Device(config)# wireless rfid
```

The default value is enabled.

Use the **no** form of this command to disable RFID tag tracking.

Step 3 Configure the RFID tag data timeout value to cleanup the table.

Example:

```
Device(config)# wireless rfid timeout timeout-value 90
```

The timeout value is the amount of time that the embedded wireless controller maintains tags before expiring them. For example, if a tag is configured to beacon every 30 seconds, we recommend that you set the timeout value to 90 seconds (approximately three times the beacon value). The default value is 1200 seconds.

Verify RFID tag support

To view the summary of RFID tags that are clients, use the command:

```
Device# show wireless rfid client
```

To view the detailed information for an RFID tag, use the command:

```
Device# show wireless rfid detail rfid-mac-address

RFID address  000c.cc96.0001
Vendor Cisco
Last Heard 6 seconds  ago
Packets Received 187
Bytes Received 226

Content Header
==============
  CCX Tag Version 0
  Tx power: 12
  Channel: 11
  Reg Class: 4
CCX Payload
==============
  Last Sequence Control 2735
  Payload length 221
  Payload Data Hex Dump:
  00000000  00 02 00 00 01 09 00 00  00 00 0c b8 ff ff ff 02  |................|
  00000010  07 42 03 20 00 00 0b b8  03 4b 00 00 00 00 00 00  |.B. .....K......|
  00000020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
  00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
  00000040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

To view the summary information for all known RFID tags, use the command:

```
Device# show wireless rfid summary

Total RFID entries: : 16
Total Unique RFID entries : 16
RFID ID VENDOR Closet AP RSSI Time Since Last Heard
0012.b80a.c791 Cisco 7069.5a63.0520 -31 3 minutes 30 seconds ago
0012.b80a.c953 Cisco 7069.5a63.0460 -33 4 minutes 5 seconds ago
0012.b80b.806c Cisco 7069.5a63.0520 -46 15 seconds ago
0012.b80d.e9f9 Cisco 7069.5a63.0460 -38 4 minutes 28 seconds ago
0012.b80d.ea03 Cisco 7069.5a63.0520 -43 4 minutes 29 seconds ago
0012.b80d.ea6b Cisco 7069.5a63.0460 -39 4 minutes 26 seconds ago
0012.b80d.ebe8 Cisco 7069.5a63.0520 -43 3 minutes 21 seconds ago
0012.b80d.ebeb Cisco 7069.5a63.0520 -43 4 minutes 28 seconds ago
0012.b80d.ec48 Cisco 7069.5a63.0460 -42 4 minutes 7 seconds ago
0012.b80d.ec55 Cisco 7069.5a63.0520 -41 1 minute 52 seconds ago
```

To view the location-based system RFID statistics, use the command:

```
Device# show wireless rfid stats

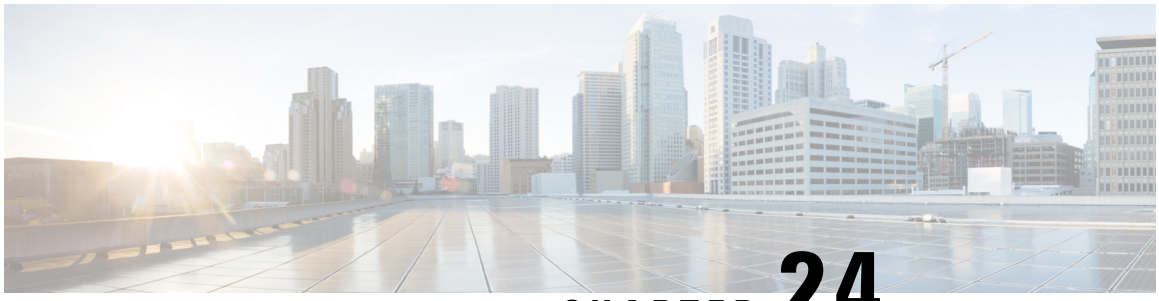
RFID stats :
==============
RFID error db full : 0
RFID error invalid paylod : 0
RFID error invalid tag : 0
RFID error dot11 hdr : 0
RFID error pkt len : 0
RFID error state drop : 0
RFID total pkt received : 369
RFID populated error value : 0
RFID error insert records : 0
RFID error update records : 0
RFID total insert record : 16
RFID ccx payload error : 0
RFID total delete record : 0
RFID error exceeded ap count : 0
```

```
RFID error record remove : 0
RFID old rssi expired count: 0
RFId smallest rssi expireed count : 0
RFID total query insert : 0
RFID error invalid rssi count : 0
```

To view the NMSP notification interval, use the command:

```
Device# show nmsp notification interval
NMSP Notification Intervals
---------------------------

RSSI Interval:
  Client              : 2 sec
  RFID                : 50 sec
  Rogue AP            : 2 sec
  Rogue Client        : 2 sec
  Spectrum            : 2 sec
```



CHAPTER 24

Application Visibility and Control

- Application visibility and control, on page 197
- AVC-based selective reanchoring, on page 205
- AVC-based selective reanchoring, on page 211
- Restrictions for AVC-based selective reanchoring, on page 212
- Configure the flow exporter (CLI), on page 212
- Configure the flow monitor (CLI), on page 212
- Configure the AVC reanchoring profile (CLI), on page 213
- Configure the wireless WLAN profile policy (CLI), on page 214
- Verify AVC reanchoring, on page 215

Application visibility and control

Application visibility and control is a wireless network feature set that

- enables real-time identification and monitoring of applications using deep packet inspection
- allows creation of policy rules to manage application bandwidth and usage, and
- integrates with Flexible NetFlow (FNF) to report traffic statistics per application or protocol.

Application visibility and control (AVC) is a subset of the Flexible NetFlow (FNF) package. AVC provides traffic information. The AVC feature uses a distributed approach that benefits from NBAR running on the AP or embedded wireless controller. The goal is to run deep packet inspection (DPI) and report the results using FNF messages.

The NBAR2 engine analyzes and recognizes traffic flows. Each specific flow is marked with the detected protocol or application. You can use this per-flow information for application visibility through FNF. After application visibility is established, you can define control rules with policing mechanisms for a client.

Guidelines for application visibility and control

- Using AVC rules, bandwidth for a particular application can be limited for all clients joined on the WLAN. These bandwidth contracts coexist with per-client downstream rate limiting. The per-client limits take precedence over the per-application rate limits.
- The FNF feature is supported in wireless deployments. It relies on NetFlow enablement on the embedded wireless controller for FlexConnect mode.

-
-
-

The behavior of the AVC solution changes based on the wireless deployment mode. The next sections describe the commonalities and differences in all scenarios.

FlexConnect Mode

- NBAR is enabled on an AP.
- AVC pushes the FNF configuration to the APs.
- Context transfer for roaming is supported in AVC-FNF.
- NetFlow exporter is supported.

Prerequisites for application visibility and control

- The APs should be AVC capable.
- Configure the application visibility feature with FNF to enable AVC control (QoS).

Restrictions for application visibility and control

- Layer 2 roaming is not supported across embedded wireless controller.
- Multicast traffic is not supported.
- AVC is supported only on the following APs:
  - Cisco Aironet 1800 Series Access Points
  - Cisco Aironet 2700 Series Access Point
  - Cisco Aironet 2800 Series Access Point
  - Cisco Aironet 3700 Series Access Points
  - Cisco Aironet 3800 Series Access Points
  - Cisco Aironet 4800 Series Access Points
- AVC is not supported on Cisco Aironet 702W, 702I (128 M memory), and 1530 Series APs.
- Only applications recognized by App Visibility can be used for applying QoS control.
- Data link is not supported for NetFlow fields in AVC.
- You cannot map the same WLAN profile to both the AVC-not-enabled policy profile and the AVC-enabled policy profile.
- NBAR-based QoS policy configuration is allowed at client level and BSSID level, configured on policy profile are not supported.

- NBAR cannot classify traffic accurately when SaaS applications use end-to-end encryption, QUIC, or DoH because encryption affects classification. In these cases, encrypted traffic including DoH and QUIC without SNI prevents NBAR from sending the correct Protocol ID, which causes traffic classification issues.

When AVC is enabled, the profile supports a maximum of 23 rules, including the default DSCP rule. If the number of rules exceeds 23, the AVC policy will not be sent to the AP.

Configure AVC

Enable monitoring and policy enforcement for application usage on the wireless network.

Procedure

Step 1 Create a flow monitor using the **record wireless avc basic** command.
Step 2 Create a wireless policy profile.
Step 3 Apply the flow monitor to the wireless policy profile.
Step 4 Create a wireless policy tag.
Step 5 Map the WLAN to the policy profile.
Step 6 Attach the policy tag to the APs.

Create a flow monitor (CLI)

The NetFlow configuration requires a flow record, a flow monitor, and a flow exporter. Make this configuration your first step in the overall AVC configuration.

Note In FlexConnect mode, the default values for **cache timeout active** and **cache timeout inactive** commands do not provide optimal AVC performance. Set both values to 60 in the flow monitor.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a flow monitor.

Example:

```
Device(config)# flow monitor monitor-name fm_avc
```

Step 3 Specify the basic wireless AVC flow template.

Example:

```
Device(config-flow-monitor)# record wireless avc basic
```

Note
The **record wireless avc basic** command is same as **record wireless avc ipv4 basic** command. However, **record wireless avc ipv4 basic** command is not supported in FlexConnect or Fabric modes. In such scenarios, use the **record wireless avc basic** command.

Configure a flow monitor (GUI)

Set up a flow monitor and export data to a collector using a pre-configured flow exporter.

Before you begin

Ensure you have already created a flow exporter.

Procedure

Step 1 Choose **Configuration** > **Services** > **Application Visibility** and go to the **Flow Monitor** tab.

Step 2 In the **Monitor** area, click **Add** to add a flow monitor.

Step 3 In the **Flow Monitor** window, add a flow monitor and a description.

Step 4 Select the Flow exporter from the drop-down list to export data from the flow monitor to the collector.

Note
To export wireless NetFlow data, use these templates:

- ETA (Encrypted Traffic Analysis)
- wireless avc basic
- wireless avc basic IPv6

Step 5 Click **Apply to Device** to save the configuration.

Create a flow exporter (CLI)

Define export parameters for a flow exporter so AVC statistics are visible on the controller embedded wireless controller. This is an optional procedure for configuring flow exporter parameters.

Note For the AVC statistics to be visible at the embedded wireless controller, you should configure a local flow exporter using these commands:

- **flow exporter** *my_local*
- **destination local wlc**

Also, your flow monitor must use this local exporter for the statistics to be visible at the embedded wireless controller.

Procedure

Step 1 Create a flow monitor.

Example:

```
Device(config)# flow exporter flow-export-name export-test
```

Step 2 Describe the flow record as a maximum 63-character string.

Example:

```
Device(config-flow-exporter)# description IPv4flow
```

Step 3 Specify the hostname or IP address of the system or the local controller to which the exporter sends data.

Example:

```
Device(config-flow-exporter)# destination {hostname/ipv4address | hostname/ipv6address |
local wlc}
```

Step 4 **Example:**

```
Device(config-flow-exporter)# destination {hostname/ipv4address | hostname/ipv6address |
local wlc}
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-flow-exporter)# end
```

Step 6 (Optional) Verify your configuration.

Example:

```
Device# show flow exporter
```

Configure a policy tag (CLI)

Establish a policy tag to control wireless device and SSID behavior.

Policy tags allow you to associate specific policy profiles with APs, enabling customized wireless network behavior.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure policy tag and enter policy tag configuration mode.

Example:

```
Device(config-policy-tag)# wireless tag policy policy-tag-name rr-xyz-policy-tag
```

Step 3 Save the configuration and exit configuration mode and return to privileged EXEC mode. **end**

Example:

```
Device(config-policy-tag)# end
```

Attach a policy profile to a WLAN interface (GUI)

Link the appropriate policy configuration to a WLAN interface for centralized control. Use the GUI to ensure new or modified policy profiles are correctly associated with WLANs before deployment.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the **Manage Tags** page, click the **Policy** tab.

Step 3 Click **Add** to open the **Add Policy Tag** window.

Step 4 Enter a name and description for the policy tag.

Step 5 Click **Add** to map WLAN and policy.

Step 6 Select the WLAN profile to map with the appropriate policy profile, and click the tick icon.

Step 7 Click **Save & Apply to Device**.

Attach a policy profile to a WLAN interface (CLI)

Assign a policy profile to a WLAN interface to control traffic and services for that WLAN.

Procedure

Step 1 Create a policy tag.

Example:

```
Device(config)# wireless tag policy avc-tag
```

Step 2 Attach a policy profile to a WLAN profile.

Example:

```
Device(config-policy-tag)# wlan wlan_avc policy avc_pol
```

What to do next

- Do not attach different AVC policy profiles to the same WLAN across different policy tags.

  This is an example of incorrect configuration:

  ```
  wireless profile policy avc_pol1
  ipv4 flow monitor fm-avc1 input
  ipv4 flow monitor fm-avc1 outputno shutdown
  wireless profile policy avc_pol2
  ipv4 flow monitor fm-avc2 input
  ipv4 flow monitor fm-avc2 output
  no shutdown
  wireless tag policy avc-tag1
  wlan wlan1 policy avc_pol1
  wireless tag policy avc-tag2
  wlan wlan1 policy avc_pol2
  ```

  This example violates the restriction stated earlier, that is, the WLAN *wlan1* is mapped to 2 policy profiles, *avc_pol1* and *avc_pol2*. This configuration is incorrect because the WLAN *wlan1* must be mapped to either *avc_pol1* or *avc_pol2* throughout the configuration..

- Conflicting policy profiles on the same WLAN are not supported. For example, policy profile (with and without AVC) applied to the same WLAN in different policy tags.

  This is an example of an incorrect configuration:

  ```
  wireless profile policy avc_pol1
  no shutdown
  wireless profile policy avc_pol2
  ipv4 flow monitor fm-avc2 input
  ipv4 flow monitor fm-avc2 output
  no shutdown
  wireless tag policy avc-tag1
  wlan wlan1 policy avc_pol1
  wireless tag policy avc-tag2
  wlan wlan1 policy avc_pol2
  ```

  In this example, a policy profile with and without AVC is applied to the same WLAN in different tags.

- Run the **no shutdown** command on the WLAN after completing the configuration.
- If the WLAN is already in **no shutdown** mode, run the **shutdown** command, then run the **no shutdown** command.

Attach a policy profile to an AP (CLI)

Assign a specific policy profile to an AP through commands.

Procedure

Step 1 Enter AP configuration mode.

Example:

```
Device(config)# ap ap-ether-mac 34a8.2ec7.4cf0
```

Step 2 Specify the policy tag that is to be attached to the AP.

Example:

```
Device(config)# policy-tag avc-tag
```

Verify the AVC configuration (CLI)

Review and validate the Application Visibility and Control (AVC) configuration on a device.

Procedure

Step 1 Display information about the top applications and the users who use them.

Example:

```
Device# show avc wlan wlan_avc top 2 applications {aggregate | downstream | upstream}
```

Note
To collect accurate statistics, ensure that wireless clients are connected to the WLAN and transmitting data. After 90 seconds, run the command.

Step 2 Display information about the top applications for the client.

Example:

```
Device# show avc client 9.3.4 top 3 applications {aggregate | downstream | upstream}
```

Note
Ensure that wireless clients are connected to the WLAN and transmitting data. Wait 90 seconds for statistics to become available, then run the command.

Step 3 Display information about the top applications and the users who uses them.

Example:

```
Device# show avc wlan wlan_avc application app top 4 aggregate
```

Step 4 Display a summary of all the APs attached to the embedded wireless controller.

Example:

```
Device# show ap summary
```

Step 5 Display a summary of all the APs with policy tags.

Example:

```
Device# show ap tag summary
```

AVC-based selective reanchoring

An AVC-based selective reanchoring is a wireless client mobility mechanism that

- reanchors clients as they roam between one embedded wireless controller to another and prevent depletion of available IP addresses, and
- uses AVC profile-based statistics to determine if a client should be reanchored or deferred.

This is useful when a client is actively running a voice or video application defined in the AVC rules.

The reanchoring process also involves deauthentication of anchored clients. The clients get deauthenticated when they do not transmit traffic for the applications listed in the AVC rules while roaming between controllers.

Restrictions for AVC-based selective reanchoring

- This feature operates only in local mode. FlexConnect and fabric modes are not supported.
- This feature does not operate in guest tunneling scenarios or export anchor scenarios.
- After reanchoring, the old IP address is retained until the IP address lease period ends.

Configure the flow exporter (CLI)

Define and configure a flow exporter to specify where and how NetFlow data is exported from the device.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a flow exporter and enter the flow exporter configuration mode.

Example:

```
Device(config)# flow exporter avc-reanchor
```

Note
You can use this command to modify an existing flow exporter too.

Step 3 Set the exporter as local.

Example:

```
Device(config-flow-exporter)# destination local wlc
```

Configure the flow monitor (CLI)

Set up a flexible NetFlow flow monitor to track traffic statistics.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a flow monitor and enter the flexible NetFlow flow monitor configuration mode.

Example:

```
Device(config)# flow monitor fm_avc
```

Note
You can use this command to modify an existing flow monitor too.

Step 3 Specify the name of an exporter.

Example:

```
Device(config-flow-monitor)# exporter avc-reanchor
```

Step 4 Specify the flow record to use to define the cache.

Example:

```
Device(config-flow-monitor)# record wireless avc basic
```

Step 5 Set the active flow timeout, in seconds.

Example:

```
Device(config-flow-monitor)# cache timeout active 60
```

Step 6 Set the inactive flow timeout, in seconds.

Example:

```
Device(config-flow-monitor)# cache timeout inactive 60
```

Configure the AVC reanchoring profile (CLI)

Configure traffic reanchoring for AVC by creating and updating an AVC reanchoring profile using commands.

Before you begin

- Ensure that you use the AVC-Reanchor-Class class map. All other class-map names are ignored by Selective Reanchoring.
- During boot up, the system checks for the existence of the AVC-Reanchor-Class class map. If it is not found, the system creates default protocols, such as jabber-video, WiFi-calling, and others. If the AVC-Reanchor-Class class map is found, configuration changes are not made. Updates to protocols saved to the startup configuration persist across reboots.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the class map.

Example:

```
Device(config)# class-map AVC-Reanchor-Class
```

Step 3 Instruct the device to match with any of the protocols that pass through it.

Example:

```
Device(config-cmap)# match any
```

Step 4 Specify a match to the application name.

Example:

```
Device(config-cmap)# match protocol jabber-audio
```

You can edit the class-map configuration later to add or remove protocols such as **jabber-video and wifi-calling** if needed.

Configure the wireless WLAN profile policy (CLI)

Follow the procedure given below to configure the WLAN profile policy.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile and enters wireless policy configuration mode and disable the policy profile.

Example:

```
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# shutdown
```

Step 3 Enable central switching.

Example:

```
Device(config-wireless-policy)# central switching
```

Step 4 Specify the name of the IPv4 ingress flow monitor.

Example:

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc input
```

Step 5 Specify the name of the IPv4 egress flow monitor.

Example:

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc output
```

Step 6 Configure a class map with protocols for the selective reanchoring feature.

Example:

```
Device(config-wireless-policy)# reanchor class AVC-Reanchor-Class
```

Step 7 Enable the policy profile.

Example:

```
Device(config-wireless-policy)# no shutdown
```

Verify AVC reanchoring

Use the commands to verify the AVC reanchoring configuration:

```
Device# show wireless profile policy detailed avc_reanchor_policy

Policy Profile Name           : avc_reanchor_policy
Description                   :
Status                        : ENABLED
VLAN                          : 1
Wireless management interface VLAN        : 34
!
.
.
.
AVC VISIBILITY                : Enabled
Flow Monitor IPv4
  Flow Monitor Ingress Name   : fm_avc
  Flow Monitor Egress Name    : fm_avc
Flow Monitor IPv6
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
NBAR Protocol Discovery       : Disabled
Reanchoring                   : Enabled
Classmap name for Reanchoring
  Reanchoring Classmap Name   : AVC-Reanchor-Class
!
.
```

```
.
.
  ----------------------------------------------------

Device# show platform software trace counter tag wstatsd chassis active R0 avc-stats debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_deassociated_clients 28340 1
Reanch_tracked_clients 28340 4
Reanch_deleted_clients 28340 3

Device# show platform software trace counter tag wncd chassis active R0 avc-afc debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_co_ignored_clients 30063 1
Reanch_co_anchored_clients 30063 5
Reanch_co_deauthed_clients 30063 4

Device# show platform software wlavc status wncd
Event history of WNCD DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.630342 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822780 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822672 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172073 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738367 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738261 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.162689 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757643 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757542 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.468749 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18857 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18717 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164304 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163877 2 :READY 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593257 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593152 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.664772 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822499 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822222 2 :READY 2 :FSM_AFM_UNBIND 0 0
```

```
06/12/2018 16:45:15.207605 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738105 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.737997 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164225 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757266 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757181 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472778 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.15413 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.15263 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164254 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163209 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163189 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- ------------------------- ---- ----
06/12/2018 16:45:30.630764 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822621 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822574 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172357 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738212 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738167 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164048 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757403 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757361 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472561 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18660 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18588 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164293 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163799 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163773 1 :INIT 24:CREATE_FSM 0 0

Device# show platform software wlavc status wncmgrd

Event history of WNCMgr DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------- ------------------------- ---- ----
06/12/2018 16:45:30.629278 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629223 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629179 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510867 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510411 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510371 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886377 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
!
```

```
AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.664032 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.663958 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.663921 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.511151 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510624 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510608 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.810867 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807239 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807205 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806734 4 :READY 3 :FSM_WLAN_DOWN 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.629414 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629392 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629380 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510954 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510572 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510532 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886293 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807844 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807795 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806990 4 :READY 3 :FSM_WLAN_DOWN 0 0
!
```

AVC-based selective reanchoring

An AVC-based selective reanchoring is a wireless client mobility mechanism that

- reanchors clients as they roam between one embedded wireless controller to another and prevent depletion of available IP addresses, and
- uses AVC profile-based statistics to determine if a client should be reanchored or deferred.

This is useful when a client is actively running a voice or video application defined in the AVC rules.

The reanchoring process also involves deauthentication of anchored clients. The clients get deauthenticated when they do not transmit traffic for the applications listed in the AVC rules while roaming between controllers.

Restrictions for AVC-based selective reanchoring

- This feature operates only in local mode. FlexConnect and fabric modes are not supported.
- This feature does not operate in guest tunneling scenarios or export anchor scenarios.
- After reanchoring, the old IP address is retained until the IP address lease period ends.

Configure the flow exporter (CLI)

Define and configure a flow exporter to specify where and how NetFlow data is exported from the device.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a flow exporter and enter the flow exporter configuration mode.

Example:

```
Device(config)# flow exporter avc-reanchor
```

Note
You can use this command to modify an existing flow exporter too.

Step 3 Set the exporter as local.

Example:

```
Device(config-flow-exporter)# destination local wlc
```

Configure the flow monitor (CLI)

Set up a flexible NetFlow flow monitor to track traffic statistics.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a flow monitor and enter the flexible NetFlow flow monitor configuration mode.

Example:

```
Device(config)# flow monitor fm_avc
```

Note
You can use this command to modify an existing flow monitor too.

Step 3 Specify the name of an exporter.

Example:

```
Device(config-flow-monitor)# exporter avc-reanchor
```

Step 4 Specify the flow record to use to define the cache.

Example:

```
Device(config-flow-monitor)# record wireless avc basic
```

Step 5 Set the active flow timeout, in seconds.

Example:

```
Device(config-flow-monitor)# cache timeout active 60
```

Step 6 Set the inactive flow timeout, in seconds.

Example:

```
Device(config-flow-monitor)# cache timeout inactive 60
```

Configure the AVC reanchoring profile (CLI)

Configure traffic reanchoring for AVC by creating and updating an AVC reanchoring profile using commands.

Before you begin

- Ensure that you use the AVC-Reanchor-Class class map. All other class-map names are ignored by Selective Reanchoring.
- During boot up, the system checks for the existence of the AVC-Reanchor-Class class map. If it is not found, the system creates default protocols, such as jabber-video, WiFi-calling, and others. If the AVC-Reanchor-Class class map is found, configuration changes are not made. Updates to protocols saved to the startup configuration persist across reboots.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the class map.

Example:

```
Device(config)# class-map AVC-Reanchor-Class
```

Step 3 Instruct the device to match with any of the protocols that pass through it.

Example:

```
Device(config-cmap)# match any
```

Step 4 Specify a match to the application name.

Example:

```
Device(config-cmap)# match protocol jabber-audio
```

You can edit the class-map configuration later to add or remove protocols such as **jabber-video and wifi-calling** if needed.

Configure the wireless WLAN profile policy (CLI)

Follow the procedure given below to configure the WLAN profile policy.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile and enters wireless policy configuration mode and disable the policy profile.

Example:

```
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# shutdown
```

Step 3 Enable central switching.

Example:

```
Device(config-wireless-policy)# central switching
```

Step 4 Specify the name of the IPv4 ingress flow monitor.

Example:

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc input
```

Step 5 Specify the name of the IPv4 egress flow monitor.

Example:

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc output
```

Step 6 Configure a class map with protocols for the selective reanchoring feature.

Example:

```
Device(config-wireless-policy)# reanchor class AVC-Reanchor-Class
```

Step 7 Enable the policy profile.

Example:

```
Device(config-wireless-policy)# no shutdown
```

Verify AVC reanchoring

Use the commands to verify the AVC reanchoring configuration:

```
Device# show wireless profile policy detailed avc_reanchor_policy

Policy Profile Name           : avc_reanchor_policy
Description                   :
Status                        : ENABLED
VLAN                          : 1
Wireless management interface VLAN        : 34
!
.
.
.
AVC VISIBILITY                : Enabled
Flow Monitor IPv4
  Flow Monitor Ingress Name   : fm_avc
  Flow Monitor Egress Name    : fm_avc
Flow Monitor IPv6
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
NBAR Protocol Discovery       : Disabled
Reanchoring                   : Enabled
Classmap name for Reanchoring
  Reanchoring Classmap Name   : AVC-Reanchor-Class
!
.
.
.
  ----------------------------------------------------

Device# show platform software trace counter tag wstatsd chassis active R0 avc-stats debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_deassociated_clients 28340 1
Reanch_tracked_clients 28340 4
Reanch_deleted_clients 28340 3

Device# show platform software trace counter tag wncd chassis active R0 avc-afc debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_co_ignored_clients 30063 1
Reanch_co_anchored_clients 30063 5
Reanch_co_deauthed_clients 30063 4

Device# show platform software wlavc status wncd
Event history of WNCD DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
```

```
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.630342 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822780 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822672 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172073 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738367 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738261 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.162689 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757643 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757542 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.468749 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18857 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18717 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164304 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163877 2 :READY 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593257 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593152 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.664772 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822499 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822222 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.207605 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738105 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.737997 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164225 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757266 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757181 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472778 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.15413 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.15263 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164254 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163209 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163189 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
```

```
06/12/2018 16:45:30.630764 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822621 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822574 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172357 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738212 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738167 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164048 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757403 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757361 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472561 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18660 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18588 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164293 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163799 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163773 1 :INIT 24:CREATE_FSM 0 0
```

```
Device# show platform software wlavc status wncmgrd

Event history of WNCMgr DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

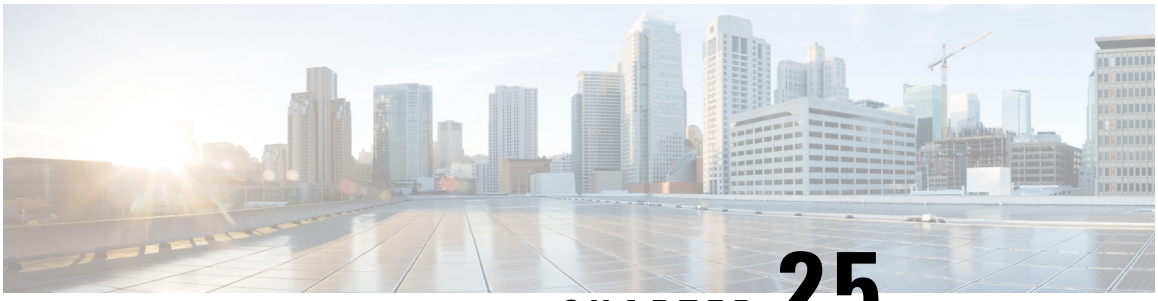
Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.629278 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629223 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629179 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510867 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510411 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510371 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886377 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.664032 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.663958 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.663921 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.511151 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510624 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510608 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.810867 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807239 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807205 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806734 4 :READY 3 :FSM_WLAN_DOWN 0 0
!
```

```
AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.629414 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629392 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629380 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510954 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510572 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510532 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886293 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807844 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807795 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806990 4 :READY 3 :FSM_WLAN_DOWN 0 0
!
```



CHAPTER 25

Cisco Spaces

A Cisco Connected Mobile Experience (CMX) deployment is a cloud-based indoor location services platform that

- provides real-time location analytics and engagement capabilities for indoor environments
- enables integration with Cisco wireless controllers through secure protocols such as HTTPS, and
- supports scalable, cloud-delivered access to location-based services.

The Network Mobility Services Protocol (NMSP) cloud-service of the wireless embedded wireless controller communicates with Cisco Spaces using HTTPS as a transport protocol.

- Configure Cisco Spaces , on page 219
- Verify Cisco Spaces configuration, on page 220

Configure Cisco Spaces

Ensure successful operation of NMSP cloud-services by configuring DNS, importing root CAs, and registering to Cisco Spaces.

To configure, Cisco Spaces, perform these procedures. These steps enable the controller to communicate securely with cloud services and Cisco Spaces.

Before you begin

- *Configure DNS*: To resolve fully qualified domain names used by NMSP cloud-services, configure a *DNS* server using the **ip name-server** *server_address* configuration command as shown in Step 2.
- *Import 3rd party root CAs*: The controller verifies the peer and host based on the certificate sent by the CMX when a connection is established. Root CA certificates are not preinstalled on the controller. Import a set of root CAs trusted by Cisco to the trustpool of the crypto PKI using the **crypto pki trustpool import url <url>** configuration command as shown in Step 3.
- Successful registration to Cisco Spaces is required before configuring the **server url** and **server token** parameters to complete setup.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the DNS on the controller to resolve the FQDN names used by the NMSP cloud-services.

Example:

```
Device(config)# ip name-server 10.10.10.205
```

Step 3 Import the 3rd party root CA.

Example:

```
Device(config)# crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b
```

The controller verifies the peer using the imported certificate.

Step 4 Configure the URL used for cloud services.

Example:

```
Device(config)# nmsp cloud-services server url https://cisco.com
```

Use the *no* form of the command to delete the server url from the configuration.

Step 5 Configure the authentication token for the NMSP cloud service.

Example:

```
Device(config)# nmsp cloud-services server token test
```

Use the *no* form of the command to delete the server token from the configuration.

Step 6 (Optional) Configure HTTP proxy details for the NMSP cloud service.

Example:

```
Device(config)# nmsp cloud-services http-proxy 10.0.0.1 port 10
```

Use the *no* form of the command to disable the use of a HTTP proxy.

Step 7 Enable NMSP cloud services.

Example:

```
Device(config)# nmsp cloud-services enable
```

Use the *no* form of the command to disable the feature.

Verify Cisco Spaces configuration

Use the commands to verify the Cisco Spaces configuration.

To view the status of active NMSP connections, use the command:

```
Device# show nmsp status

MSE IP Address   Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data   Transport
------------------------------------------------------------------------------
192.0.2.1        0             0             1           1         TLS
198.51.100.1     0             0             1230        2391      HTTPs
```

To view the NMSP cloud service status, use the command:

```
Device# show nmsp cloud-services summary

CMX Cloud-Services Status
-------------------------
Server:                                 https://yenth8.cmxcisco.com
IP Address:                             198.51.100.1
Cmx Service:                            Enabled
Connectivity:                           https: UP
Service Status:                         Active
Last Request Status:                    HTTP/1.1 200 OK
Heartbeat Status:                       OK
```

To view the NMSP cloud service statistics, use the command:

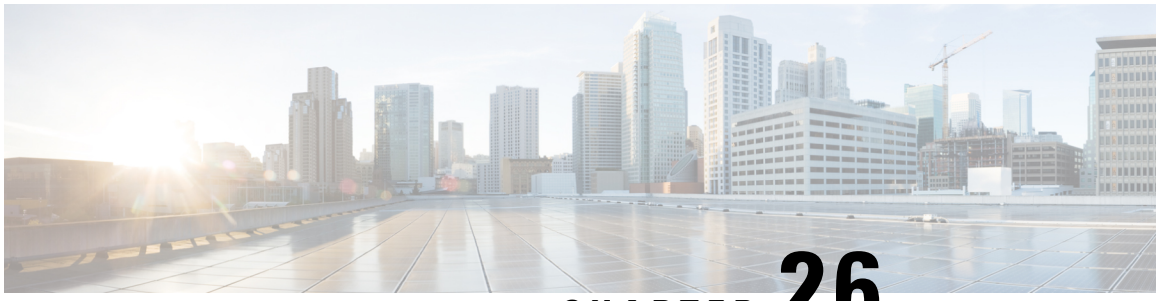
```
Device# show nmsp cloud-services statistics

CMX Cloud-Services Statistics
-----------------------------
Tx DataFrames:                           3213
Rx DataFrames:                           1606
Tx HeartBeat Req:                       31785
Heartbeat Timeout:                          0
Rx Subscr Req:                           2868
Tx DataBytes:                           10069
Rx DataBytes:                           37752
Tx HeartBeat Fail:                          2
Tx Data Fail:                               0
Tx Conn Fail:                               0
```

To view the mobility services summary, use the command:

```
Device# show nmsp subscription summary

Mobility Services Subscribed:
Index Server IP Services
----- --------- --------
1 209.165.200.224 RSSI, Info, Statistics, AP Monitor, AP Info
2 209.165.200.224 RSSI, Statistics, AP Info
```



CHAPTER 26

EDCA Parameters

- EDCA parameters, on page 223

EDCA parameters

An Enhanced Distributed Channel Access (EDCA) parameter is a wireless channel access configuration that

- assigns preferential access to voice, video, and other quality of service (QoS) traffic
- differentiates wireless traffic priority through distinct parameter settings, and
- enables customization of wireless performance using controller GUI or CLI interfaces.

Configure EDCA parameters (GUI)

Set global EDCA parameters to optimize channel access for voice, video, and quality-of-service traffic on your wireless networks using GUI.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **Parameters**. On this window, you can configure global parameters for 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios.

Note
You cannot configure or modify parameters, if the radio network is enabled. Disable the network status in the **Configuration** > **Radio Configurations** > **Network** window before proceeding. To apply the EDCA settings to the WLANs, disable and then re-enable the WLANs.

Step 2 In the **EDCA Parameters** section, select an EDCA profile from the **EDCA Profile** drop-down list. Enhanced Distributed Channel Access (EDCA) parameters provide preferential wireless channel access for voice, video, and other QoS traffic.

Step 3 Click **Apply**.

Configure EDCA parameters (CLI)

Set EDCA parameters on a wireless controller to optimize Quality of Service (QoS) for different traffic profiles and improve network performance using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Disable the radio network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} shutdown
```

Step 3 Enable specific EDCA parameters for the 802.11a or 802.11b/g network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | fastlane |
optimized-video-voice | optimized-voice | svp-voice | wmm-default}
```

- **custom-voice**: Enables custom voice parameters for the 802.11a or 802.11b/g network.
- **fastlane**: Enables the fastlane parameters for the 802.11a or 802.11b/g network.
- **optimized-video-voice**: Enables EDCA voice-optimized and video-optimized parameters for the 802.11a or 802.11b/g network. Choose this option when both voice and video services are deployed on your network.
- **optimized-voice**: Enables non-SpectraLink voice-optimized profile parameters for the 802.11a or 802.11b/g network. Choose this option when voice services other than SpectraLink are deployed on your network.
- **svp-voice**: Enables SpectraLink voice-priority parameters for the 802.11a or 802.11b/g network. Choose this option if SpectraLink phones are deployed on your network to improve the quality of calls.
- **wmm-default**: Enables the Wi-Fi Multimedia (WMM) default parameters for the 802.11a or 802.11b/g network. This is the default option. Choose this option when voice or video services are not deployed on your network.

Step 4 Re-enable the radio network.

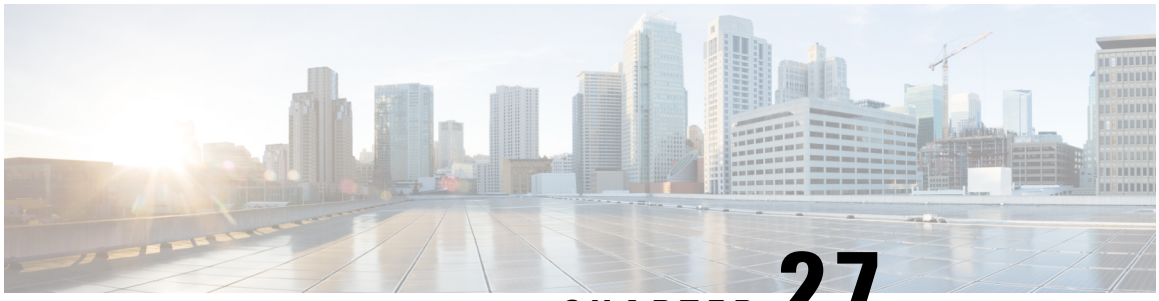
Example:

```
Device(config)# no ap dot11 {5ghz | 24ghz} shutdown
```

Step 5 Display the current status of MAC optimization for voice.

Example:

```
Device# show ap dot11 {5ghz | 24ghz} network
```



CHAPTER 27

802.11 parameters and Band Selection

- Band select, on page 225

Band select

Band selection is a WLAN feature that

- enables dual-band client radios to connect to less congested 5-GHz APs
- improves overall network performance by reducing interference experienced on the controller, and
- uses probe response regulation to make 5-GHz channels more attractive to clients by delaying probe responses to clients on 2.4-GHz channels.

The 2.4-GHz band is often congested. Clients using the 2.4-GHz band often experience interference from Bluetooth devices, microwave ovens, cordless phones, and other access points. The 802.11b/g limit of three nonoverlapping channels also causes co-channel interference.

Band select works by regulating probe responses to clients and can be enabled on a per-WLAN basis. To view the band select table on an AP, run the **show dot11 band-select** command. It can also be viewed by running the **show cont d0/d1 | begin Lru** command.

Band Select Algorithm

The band select algorithm affects clients that use 2.4-GHz band. When a client sends a probe request to an access point, the client's probe Active and Count values become one as seen in the band select table. The algorithm functions based on the following scenarios:

- Scenario 1: The client RSSI (as seen from the show cont d0/d1 | begin RSSIcommand output) is greater than both Mid RSSI and Acceptable Client RSSI.
  1. For dual-band clients, the access point does not send 2.4-GHz probe responses but sends 5-GHz probe responses for all 5-GHz probe requests.
  2. For single-band (2.4-GHz) clients, the access point sends 2.4-GHz probe responses only after the probe suppression cycle.
  3. When the client's probe count matches the configured probe cycle count, the algorithm waits for the Age Out Suppression time, marks the client probe's Active value as zero, and restarts.

- Scenario 2: Client RSSI (as seen from show cont d0/d1 | begin RSSI) lies between Mid-RSSI and Acceptable Client RSSI.

  1. All 2.4-GHz and 5-GHz probe requests are responded to without any restrictions.
  2. This scenario is similar to when band select is disabled.

     Note: The client RSSI value (from the sh cont d0 | begin RSSI command output) averages all client packets received. The Mid RSSI feature reports the instantaneous RSSI value of probe packets. Therefore, the client RSSI often appears weaker than the configured Mid RSSI value (by 7 dB). The 802.11b probes from the client are suppressed to push the client to associate with the 802.11a band.

802.11 bands

Configure the 802.11b/g/n (2.4 GHz) and 802.11a/n (5 GHz) on the controller to meet your country's regulatory requirements. By default, both 802.11b/g/n and 802.11a/n are enabled.

- 802.11n Parameters
- 802.11h Parameters

802.11n Parameters

This section provides instructions for managing 802.11n APs on your network. The 802.11n devices support the 2.4 and 5-GHz bands and offer high throughput data rates.

The 802.11n high throughput rates are available on all the 802.11n APs for the WLANs using WMM with no Layer 2 encryption or with WPA2/AES encryption enabled.

Note If you want to disable MCS rates for 802.11n, 802.11ac, or 802.11ax, make sure at least one MCS rate remains enabled. To use only legacy 802.11a/b/g rates on your APs, first disable 802.11ax and 802.11ac on the controller for the specific band, then disable 802.11n. When you disable 802.11n globally on the controller, it affects all APs, even those mapped to a Custom RF Profile.

802.11h Parameters

802.11h informs client devices about channel changes and can limit the transmit power for those devices.

Restrictions for band selection, 802.11 bands, and parameters

- Band selection-enabled WLANs do not support time-sensitive applications, such as voice and video, because roaming delays impact performance.
- Band selection is supported only on Cisco Wave 2 and 802.11ax APs.

  For more specific APs, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html .
- Band selection works only when your AP is connected to a controller. If you use a FlexConnect AP without a controller connection, it does not perform band selection after a reboot.

- Dual-band clients are directed from the 2.4-GHz radio to the 5-GHz radio of the same AP, but only when both radios are operational.
- To enable or disable band selection and client load balancing, configure these features for a specific WLAN. These settings are enabled globally by default, and you cannot change them globally through the controller GUI or CLI.

Configure band selection (GUI)

Before you begin

Ensure that you have configured an AP Join profile prior to configuring the primary and backup controllers.

Optimize wireless client performance by controlling band selection settings on the controller using GUI.

Procedure

Step 1 Choose **Configuration** > **Wireless Advanced** > **Band Select**.

Step 2 In the **Cycle Count** field, enter a value between one and 10. The cycle count sets the number of suppression cycles for a new client. The default cycle count is two.

Step 3 In the **Cycle Threshold (milliseconds)** field, enter a value between one and 1000 milliseconds for the scan cycle period threshold. This setting determines the time threshold during which new probe requests from a client come from a new scanning cycle. The default cycle threshold is 200 milliseconds.

Step 4 In the **Age Out Suppression (seconds)** field, enter a value between 10 and 200 seconds. Age-out suppression sets the expiration time for pruning previously known 802.11b/g/n clients. The default value is 20 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 5 In the **Age Out Dual Band (seconds)** field, enter a value between 10 and 300 seconds. The age-out period sets the expiration time for pruning previously known dual-band clients. The default value is 50 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 6 In the Client RSSI (dbm) field, enter a value between -90 to -20. This is the average of the client packets received.

Step 7 In the Client Mid RSSI (dbm) field, enter a value between -90 to -20. This is the instantaneous RSSI value of the probe packets.

Step 8 On the **AP Join Profile** page, click the AP Join Profile name.

Step 9 Click **Apply**.

Configure band selection (CLI)

Enable and fine-tune band selection parameters for wireless clients to optimize network performance using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Set the probe cycle count for band select.

Example:

```
Device(config)# wireless client band-select cycle-count 3
```

The valid range is between one and 10.

Step 3 Set the time threshold for a new scanning cycle period.

Example:

```
Device(config)# wireless client band-select cycle-threshold 5000
```

The valid range is between one and 1000.

Step 4 Set the suppression expire to the band select.

Example:

```
Device(config)# wireless client band-select expire suppression 100
```

The valid range is between 10 and 200.

Step 5 Set the dual band expire.

Example:

```
Device(config)# wireless client band-select expire dual-band 100
```

The valid range is between 10 and 300.

Step 6 Set the client RSSI threshold.

Example:

```
Device(config)# wireless client band-select client-rssi 40
```

The valid range is between 20 and 90.

Step 7 Configure band selection on specific WLANs.

Example:

```
Device(config)# wlan wlan1 25 ssid12
Device(config-wlan)# band-select
```

The valid range is between one and 512. You can specify a value for the *SSID_network_name* parameter using up to 32 alphanumeric characters.

Configure the 802.11 bands (GUI)

Modify wireless radio settings to optimize network performance for 2.4 GHz and 5 GHz bands using commands.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **Network**.

Step 2 Click on either **5 GHz Band** or **2.4 GHz Band**. Uncheck the **Network Status** check box to disable the network. This allows configuration of the network parameters.

Step 3 In the **Beacon Interval** field, enter the rate at which the SSID is broadcast by the APs, from 100 to 600 milliseconds. The default is 100 milliseconds. For 802.11b/g/n (2.4-GHz) radios, to enable short preamble on the radio, check the **Short Preamble** check box. A short preamble improves throughput performance.

Step 4 In the **Fragmentation Threshold (in bytes)** field, enter a value between 256 to 2346 bytes. Packets larger than the size you specify here will be fragmented.

Step 5 Check the **DTPC Support** check box to advertise the transmit power level of the radio in the beacons and the probe responses. Client devices using dynamic transmit power control (DTPC) receive channel and power level information from APs. They then adjust their settings automatically. For example, a client device used primarily in Japan could rely on DTPC to adjust its channel and power settings automatically. When it travels to Italy and joins a network there, the settings are updated as needed. If the **DTPC Support** check box is checked, you cannot configure a power constraint value on your 802.11a/n/ac (5-GHz) radio network. Click **Apply**.

Step 6 In the **CCX Location Measurement** section, check the **Mode** check box to globally enable CCX radio management for the network. Once enabled, APs connected to this device issue broadcast radio measurement requests to clients running CCX v2 or later releases.

Step 7 In the **Interval** field, enter a value to specify how often the APs must issue broadcast radio measurement requests. Click **Apply**.

Step 8 In the **Data Rates** section, choose a value to specify the rates at which data can be transmitted between the AP and the client:

- Mandatory: Clients must support this data rate to associate with an AP on the controllerembedded wireless controller.
- Supported: Any associated client supporting this data rate may communicate with the AP using that rate.
- Disabled: Clients specify the data rates used for communication.

Step 9 Click **Apply** and save the configuration.

Configure the 802.11 bands (CLI)

Configure the various 802.11 wireless bands and essential parameters using commands.

Follow the procedure given below to configure 802.11 bands and parameters:

Procedure

Step 1 Disable the 802.11a and 802.11b bands.

Example:

```
Device(config)# ap dot11 5ghz shutdown
```

```
Device(config)# ap dot11 24ghz shutdown
```

Note
You must disable the 802.11a band before configuring the 802.11a network parameters.

You must disable the 802.11b band before configuring the 802.11b network parameters.

Step 2 Specify the rate at which the SSID is broadcast by the corresponding AP and the size at which packets are fragmented.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz | } beaconperiod 500
Device(config)# ap dot11 {5ghz | 24ghz | } fragmentation 300
```

The beacon interval is measured in time units (TUs). One TU is 1024 microseconds. You can configure the AP to send a beacon every 20 to 1000 milliseconds.

The threshold is a value between 256 and 2346 bytes (inclusive). Specify a low number for areas where communication is poor or where there is a great deal of radio interference.

Step 3 Enable APs to advertise their channels and transmit the power levels in beacons and probe responses.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz | } dtpc
Device(config)# no ap dot11 {5ghz | 24ghz | } dtpc
```

The default value is enabled. Client devices using dynamic transmit power control (DTPC) receive the channel-level and power-level information from the APs and adjust their settings automatically. For example, a client device used primarily in Japan can rely on DTPC to adjust its channel and power settings automatically when it travels to Italy and joins a network there.

The **no** form of the command disables the DTPC setting.

Step 4 Specify the maximum allowed number of clients that can be configured.

Example:

```
Device(config)# wireless client association limit number 50 interval milliseconds 1000
```

You can configure the maximum number of association requests on a single access point slot at a given interval. The range of association limit that you can configure is from one to 100.

The association request limit interval is measured between 100 to 10000 milliseconds.

Step 5 Specify the rate at which data can be transmitted between the controllerembedded wireless controller and the client.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} rate 36 {disable | mandatory | supported}
```

- **disable**: Defines that the clients specify the data rates used for communication.
- **mandatory**: Defines that the clients support this data rate in order to associate to an AP on the controllerembedded wireless controller.
- **supported**: Any associated clients that support this data rate can communicate with the AP using that rate. However, the clients are not required to use this rate in order to associate.
- *rate*: Specifies the rate at which data is transmitted. For the 802.11a and 802.11b bands, the data is transmitted at the rate of one, two, 5.5, six, nine, 11, 12, 18, 24, 36, 48, or 54 Mbps.

Step 6 Enable the 802.11a and 802.11b bands.

Example:

```
Device(config)# no ap dot11 5ghz shutdown
Device(config)# no ap dot11 24ghz shutdown
```

Note
The default value is enabled.

Step 7 Enable or disable 802.11g network support. Then return to privileged EXEC mode.

Example:

```
Device(config)# ap dot11 24ghz dot11g
Device(config)# end
```

The default value is enabled. You can use this command only if the 802.11b band is enabled. If you disable this feature, the 802.11b band is enabled without 802.11g support.

Configure a band-select RF profile (GUI)

Configure specific parameters in a band-select RF profile to optimize client band steering behavior on your wireless network.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Advanced**.

Step 2 In the **Band Select** tab, enter a value between one and 10 in the **Cycle Count** field. The cycle count determines how many suppression cycles are applied to each new client. The default cycle count is two.

Step 3 In the **Cycle Threshold** field, enter a value between one and 1000 milliseconds to define the scan cycle period threshold. his setting specifies the time period in which probe requests from a client are recognized as part of a new scanning cycle. The default cycle threshold value is 200 milliseconds.

Step 4 In the **Age Out Suppression** field, enter a value between 10 and 200 seconds. Age-out suppression establishes the expiration time for removing records of previously known 802.11b/g/n clients. The default value is 20 seconds. After the expiration time, clients are treated as new and become subject to probe response suppression.

Step 5 In the **Age Out Dual Band** field, enter a value between 10 and 300 seconds. The age-out period establishes the expiration time for removing records of previously known dual-band clients. The default value is 50 seconds. After the expiration period, clients are considered new and are subject to probe response suppression.

Step 6 In the **Client RSSI** field, enter a value between -90 dBm and -20 dBm. This value sets the minimum RSSI required for a client to respond to a probe.

Step 7 In the **Client Mid RSSI** field, enter a value between –20 dBm and –90 dBm. This parameter defines the mid-RSSI value, which can be used to toggle 2.4 GHz probe suppression depending on the RSSI.

Step 8 Click **Apply**.

Configure 802.11n parameters (GUI)

Apply custom 802.11n wireless settings to an RF profile in the device management GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **RF**.

Step 2 Click **Add** to view the **Add RF Profile** window.

Step 3 In the **802.11** tab, complete these steps:

a) Choose the required operational rates.

b) Select the required **802.11n MCS Rates** by checking the corresponding check boxes.

Step 4 Click **Save & Apply to Device**.

Configure 802.11n parameters (CLI)

Configure and optimize 802.11n settings on a wireless AP using commands.

Procedure

Step 1 Enable 802.11n support on the network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n
```

The **no** form of this command disables the 802.11n support on the network.

Step 2 Specify the modulation and coding scheme (MCS) rates for data transmission between the access point and the client.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n mcs tx 20
```

rtu: The valid range is zero to 23.

The **no** form of this command disables the MCS rates that are configured.

Step 3 Enable WMM on the WLAN and uses the 802.11n data rates that you configured.

Example:

```
Device(config)# wlan wlan1 25 ssid12
Device(config-wlan)# wmm require
```

The **require** keyword requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.

Step 4 Disable the network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} shutdown
```

Step 5 Specify the aggregation method used for 802.11n packets.

Example:

```
Device(config)# {ap | no ap} dot11 {5ghz | 24ghz} dot11n a-mpdu tx priority {all | 0-7}
```

Aggregation is the process of grouping packet data frames together, rather than transmitting them separately. Two aggregation methods are available: Aggregated MAC Protocol Data Unit (A-MPDU) and Aggregated MAC Service Data Unit (A-MSDU). Both A-MPDU and A-MSDU are performed in the software.

You can specify the aggregation method for various types of traffic from the access point to the clients.

The list defines the priority levels (zero-7) assigned per traffic type.

- Zero—Best effort
- One—Background
- Two—Spare
- Three—Excellent effort
- Four—Controlled load
- Five—Video, less than 100-ms latency and jitter
- Six—Voice, less than 100-ms latency and jitter
- Seven—Network control

You can configure each priority level independently or use all the parameters to configure every priority level simultaneously. You can configure priority levels so that the traffic uses either A-MPDU transmission or A-MSDU transmission.

- When you use the **ap** command with other options, traffic at that priority level uses A-MPDU transmission.
- When you use the **no ap** command along with the other options, the traffic associated with that priority level uses A-MSDU transmission.

  Configure the priority levels to match the aggregation method used by the clients. By default, both A-MPDU and A-MSDU are enabled for priority levels zero to five; all other levels are disabled.

Note
Tracking the TX retries value by internal MSDUs instead of the whole PPDU in Cisco IOS XE 17.12 and later versions provides a more accurate count for the AP's retries. This adjustment has led to a notable increase in retry count values because multiple MSDUs are contained within a single PPDU. This change is intentional and will be implemented in all future versions of Cisco IOS XE.

Step 6 Re-enable the network.

Example:

```
Device(config)# no ap dot11 {5ghz | 24ghz} shutdown
```

Step 7 Configure the guard interval for the network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}
```

Step 8 Configure the Reduced Interframe Space (RIFS) for the network.

Example:

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n rifs rx
```

Step 9 Return to privileged EXEC mode. Alternatively, press **Ctrl-Z** to exit global configuration mode.

Example:

```
Device(config)# end
```

Configure 802.11h parameters (CLI)

Configure 802.11h parameters on a wireless AP using commands to manage compliance, interference, and transmit power settings.

Procedure

Step 1 Disable the 802.11 network.

Example:

```
Device(config)# ap dot11 5ghz shutdown
```

Step 2 Enable or disable the AP to announce when it is switching to a new channel.

Example:

```
Device(config)# {ap | no ap} dot11 5ghz channelswitch mode 0
```

switch_mode: Enter zero or one to specify whether transmissions are restricted until the actual channel switch (***zero***) or are not restricted (***one***). The default value is disabled.

Step 3 Configure the 802.11h power constraint value in dB.

Example:

```
Device(config)# ap dot11 5ghz power-constraint 200
```

The valid range is from zero to 255. The default value is three.

Step 4 Re-enable the 802.11a network.

Example:

```
Device(config)# no ap dot11 5ghz shutdown
```

Verify configuration settings using band selection and 802.11 bands commands

Use these commands to verify band selection, 802.11 bands, and parameters on the device.

Table 12: Monitoring configuration settings using band selection and 802.11 band commands

Command	Purpose

show ap dot11 5ghz network	Displays 802.11a band network parameters, 802.11a operational rates, 802.11n MCS settings, and 802.11n status information.
show ap dot11 24ghz network	Displays 802.11b band network parameters, 802.11b/g operational rates, 802.11n MCS settings, and 802.11n status information.
show wireless dot11h	Displays 802.11h configuration parameters.
show wireless band-select	Displays band-select configuration settings.

Example: view the configuration settings for the 5-GHz band

```
Device# show ap dot11 5ghz network

802.11a Network : Enabled
11nSupport : Enabled
  802.11a Low Band : Enabled
  802.11a Mid Band : Enabled
  802.11a High Band : Enabled
802.11a Operational Rates
  802.11a 6M : Mandatory
  802.11a 9M : Supported
  802.11a 12M : Mandatory
  802.11a 18M : Supported
  802.11a 24M : Mandatory
  802.11a 36M : Supported
  802.11a 48M : Supported
  802.11a 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
```

```
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 36
Default Tx Power Level : 1
DTPC Status : Enabled
Fragmentation Threshold : 2346
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
TI Threshold : 0
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type check : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwidth sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: view the configuration settings for the 2.4-GHz band

```
Device# show ap dot11 24ghz network

802.11b Network : Enabled
11gSupport : Enabled
11nSupport : Enabled
  802.11b/g Operational Rates
  802.11b 1M : Mandatory
  802.11b 2M : Mandatory
  802.11b 5.5M : Mandatory
  802.11g 6M : Supported
  802.11g 9M : Supported
```

```
    802.11b 11M : Mandatory
    802.11g 12M : Supported
    802.11g 18M : Supported
    802.11g 24M : Supported
    802.11g 36M : Supported
    802.11g 48M : Supported
    802.11g 54M : Supported
  802.11n MCS Settings:
    MCS 0 : Supported
    MCS 1 : Supported
    MCS 2 : Supported
    MCS 3 : Supported
    MCS 4 : Supported
    MCS 5 : Supported
    MCS 6 : Supported
    MCS 7 : Supported
    MCS 8 : Supported
    MCS 9 : Supported
    MCS 10 : Supported
    MCS 11 : Supported
    MCS 12 : Supported
    MCS 13 : Supported
    MCS 14 : Supported
    MCS 15 : Supported
    MCS 16 : Supported
    MCS 17 : Supported
    MCS 18 : Supported
    MCS 19 : Supported
    MCS 20 : Supported
    MCS 21 : Supported
    MCS 22 : Supported
    MCS 23 : Supported
  802.11n Status:
    A-MPDU Tx:
      Priority 0 : Enabled
      Priority 1 : Disabled
      Priority 2 : Disabled
      Priority 3 : Disabled
      Priority 4 : Enabled
      Priority 5 : Enabled
      Priority 6 : Disabled
      Priority 7 : Disabled
    A-MSDU Tx:
      Priority 0 : Enabled
      Priority 1 : Enabled
      Priority 2 : Enabled
      Priority 3 : Enabled
      Priority 4 : Enabled
      Priority 5 : Enabled
    Priority 6 : Disabled
      Priority 7 : Disabled
    Guard Interval : Any
    Rifs Rx : Enabled
  Beacon Interval : 100
  CF Pollable Mandatory : Disabled
  CF Poll Request Mandatory : Disabled
  CFP Period : 4
  CFP Maximum Duration : 60
  Default Channel : 11
  Default Tx Power Level : 1
  DTPC Status : true
  Call Admission Limit : 105
  G711 CU Quantum : 15
  ED Threshold : -50
```

```
Fragmentation Threshold : 2346
PBCC Mandatory : Disabled
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
RTS Threshold : 2347
Short Preamble Mandatory : Enabled
Short Retry Limit : 7
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwidth sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: view the status of 802.11h parameters

```
Device# show wireless dot11h

Power Constraint: 0
Channel Switch: 0
Channel Switch Mode: 0
```

Example: verify the band-selection settings

This example displays a band-select configuration:

```
Device# show wireless band-select

Band Select Probe Response   : per WLAN enabling
Cycle Count                  : 2
Cycle Threshold (millisec)   : 200
Age Out Suppression (sec)    : 20
Age Out Dual Band (sec)      : 60
Client RSSI (dBm)            : -80
Client Mid RSSI (dBm)        : -80
```

Examples: band selection configuration

This example shows how to set the probe cycle count and time threshold for a new scanning cycle period for band select:

```
Device# configure terminal
(config)# wireless client band-select cycle-count 3
```

```
(config)# wireless client band-select cycle-threshold 5000
(config)# end
```

This example shows how to set the suppression expiry time to the band select:

```
Device# configure terminal
(config)# wireless client band-select expire suppression 100
(config)# end
```

This example shows how to set the dual-band expiry time for the band select:

```
Device# configure terminal
(config)# wireless client band-select expire dual-band 100
(config)# end
```

This example shows how to set the client RSSI threshold for the band select:

```
Device# configure terminal
(config)# wireless client band-select client-rssi 40
(config)# end
```

This example shows how to configure band selection on specific WLANs:

```
Device# configure terminal
(config)# wlan wlan1 25 ssid12
(config-wlan)# band-select
(config)# end
```

Examples: 802.11 bands configuration

This example shows how to configure 802.11 bands using beacon interval, fragmentation, and dynamic transmit power control:

```
Device# configure terminal
config)# ap dot11 5ghz shutdown
(config)# ap dot11 24ghz shutdown
(config)# ap dot11 5ghz beaconperiod 500
(config)# ap dot11 5ghz fragmentation 300
(config)# ap dot11 5ghz dtpc
(config)# wireless client association limit 50 interval 1000
(config)# ap dot11 5ghz rate 36 mandatory
(config)# no ap dot11 5ghz shutdown
(config)# no ap dot11 24ghz shutdown
(config)# ap dot11 24ghz dot11g
(config)# end
```

Examples: 802.11n configuration

This example shows how to configure 802.11n parameters for 5-GHz band using aggregation method:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n a-mpdu tx priority all
(config)# no ap dot11 5ghz shutdown
(config)# exit
```

This example shows how to configure the guard interval for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# no ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n guard-interval long
(config)# end
```

This example shows how to configure the RIFS for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n rifs rx
(config)# end
```

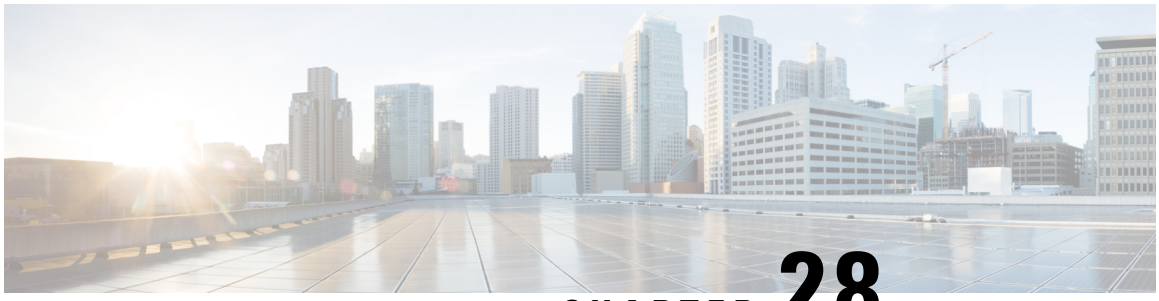
Examples: 802.11h configuration

This example shows how to configure the AP to announce when it is switching to a new channel using restriction transmission:

```
Device# configure terminal
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz channelswitch mode 0
(config)# no ap dot11 5ghz shutdown
(config)# end
```

This example shows how to configure the 802.11h power constraint for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz power-constraint 200
(config)# no ap dot11 5ghz shutdown
(config)# end
```



CHAPTER 28

Image Download

- Information About Image Download, on page 241
- Prerequisites for Image Download, on page 244
- Configuring Image Download Profile, on page 245
- Initiating Pre-Download (CLI), on page 249
- Verifying Image Download, on page 250

Information About Image Download

Software updates ensure that all the access points in the Cisco Embedded Wireless Controller network are running the latest software. The software update or image download can be performed using both the GUI and the CLI.

A typical Cisco Embedded Wireless Controller network contains the following components:

- Cisco Catalyst APs acting as controller (embedded wireless controller)
- Cisco Embedded Wireless Controller-capable APs (Other Cisco Catalyst series APs that participate in the Virtual Router Redundancy Protocol (VRRP)-based election process)
- Subordinate APs (Cisco Catalyst Series or Cisco Aironet Series Wave 2 APs)
- External TFTP and SFTP server.

Note For best user experience when using the GUI, view the browser at 100% resolution. The lines may break if the resolution is greater than 100%.

Updates to the AP Image Predownload Status (GUI)

From Cisco IOS XE Amsterdam, Release 17.3.1 onwards, during an access point (AP) image download, the Cisco Embedded Wireless Controller on Catalyst Access Points calculates the current percentage of the download and the estimated completion time of the download. (You can view these values in the CLI output by running the **show wireless ewc-ap ap image predownload status** command.)

To access the **Software Upgrage** window, from the Cisco Embedded Wireless Controller on Catalyst Access Points home page, choose **Administration** > **Software Management** > **Software Upgrade**.

The **Software Update Status** section in the GUI displays the update status bar that shows the progress of a software update, such as, **Initiate**, **Controller Image Download**, **AP Image Download**, **Network Upgrade**, **Activate**, and **Reload**.

To view the logs, click the **Show Install Logs** link.

The **Status** field displays the current status of the upgrade and indicates further action, if any, that you should perform.

The other details displayed in the window are - **Total Number of APs**, **Initiated**, **Predownloading AP Image**, **Predownloading Controller Image**, **Completed Predownloading AP Image**, **Completed Predownloading Controller Image**, **Failed to Predownload AP Image**, **Failed to Predownload Controller Image**.

The currently active AP, the AP on standby, and the preferred active AP are also displayed.

Image Download Scenarios

In a Cisco Embedded Wireless Controller network, image download from the embedded wireless controller to the subordinate AP takes place in the following scenarios:

- During AP join
- During network software upgrade (pre-download)

Image Download During AP Join

APs with older software trying to join the Cisco Embedded Wireless Controller network are automatically upgraded to match the latest software version on the embedded wireless controller. The embedded wireless controller compares the software version on the new AP with that on itself. If there is a mismatch, the AP requests the controller for a software upgrade and image download is triggered. The embedded wireless controller facilitates the transfer of the latest software from an external TFTP server or SFTP server, to the new AP.

Depending on the new AP joining the network, there are two image downloads that take place:

- **AP software image download:** This applies to all new APs joining the Cisco Embedded Wireless Controller.
- **Controller software image download:** This applies only to Cisco Catalyst series APs, capable of becoming a controller, trying to join the Cisco Embedded Wireless Controller network.

AP Software Image Download

Any Cisco Catalyst Series AP or Cisco Aironet Series Wave 2 AP can only join an embedded wireless controller if its AP software image version matches that of the controller.

During the AP join process, the embedded wireless controller first checks the AP software image version on the new AP and if it does not match what is on the controller, the latest AP software is downloaded from the controller to the new AP. Once the AP software image on the new AP is upgraded to match the version that is on the embedded wireless controller in the network, the new AP reloads. Once the new AP is back up with the upgraded AP software image, it joins the embedded wireless controller.

Controller Software Image Download

If the new AP joining the network is a CiscoCatalyst Series AP capable of becoming an embedded wireless controller, first the controller checks the AP software image on the new AP and if outdated, it is upgraded to

match the AP software version on the controller. The AP then reloads with the new AP software image and joins theembedded wireless controller in the network.

Next, the embedded wireless controller does a similar check to compare the controller software version on the embedded wireless controller-capable AP. Similar to the AP software upgrade, if there is a mismatch, the controller software on this CiscoCatalyst Series AP is also upgraded to the latest version on the embedded wireless controller. The AP reloads again, this time with the upgraded controller software image.

Efficient AP Join

If the Cisco Embedded Wireless Controllerr network contains an AP of the same image type as the newly joining AP, then the new AP downloads the AP software image from this AP. For example, if a CiscoCatalyst 9130AX Series AP is newly joining the Cisco Embedded Wireless Controller network and another CiscoCatalyst 9130AX Series AP already exists in the network, then the new AP gets its AP software image from the already joined AP.

This method, known as efficient AP join, enables homogenous APs to get the software locally (within the Cisco Embedded Wireless Controller network) rather than downloading it from an external server. This improves software download efficiency.

The first AP of a series that joins the network and downloads the software from the embedded wireless controller is called a primary image. The other APs of the same series are known as image subordinates.

Network Software Upgrade (Pre-Download)

In the pre-download scenario, image download in the Cisco Embedded Wireless Controller network occurs to upgrade the software on all the APs from one software version to another. However, these APs continue to serve existing as well as new clients and there is no network disruption.

For pre-download, all the APs should be connected to the embedded wireless controller in a stable join state. Once image download is initiated during pre-download, new APs are not allowed to join the embedded wireless controller.

Efficient AP Upgrade

In this method, the first AP of an AP series to get the image from the embedded wireless controller becomes the primary image. The remaining APs of the same AP series, the image subordinates, then download the software image locally from this primary image. This method is also known as efficient AP upgrade.

Methods Supported for Image Download

In a Cisco Embedded Wireless Controller network, there are four ways in which the software image can be downloaded from the embedded wireless controller. These methods are based on the location from where the controller transfers the software image to the subordinate AP:

- From an external TFTP server
- From an external SFTP server
- From the desktop (via HTTP)

TFTP Image Download Method

In the TFTP method, the AP and controller software images are stored on a TFTP server. To download the software images from the TFTP server, you need to specify the IP address of the TFTP server and the path to the software image bundle on the TFTP server.

The TFTP image download method can be triggered using both the GUI and CLI.

SFTP Image Download Method

In the SFTP method, the AP and controller software images are stored on an SFTP server. To download the software images from the SFTP server, in addition to the IP address of the SFTP server and the software image bundle path, you need to specify the SFTP server credentials.

The SFTP image download method also can be triggered using both the GUI and CLI.

Desktop (HTTP) Image Download Method

Image download through desktop (HTTP) is applicable only in the network software upgrade (pre-download) scenario.

For the desktop (HTTP) method, download the software image bundle for the Cisco Embedded Wireless Controller to your computer or laptop desktop. This downloaded bundle contains the AP and controller software images which need to be extracted to the computer or laptop desktop before they can be uploaded to the embedded wireless controller.

Note that the desktop (HTTP) method works only for a homogenous network. A homogenous Cisco Embedded Wireless Controller network is one which contains APs that have the same AP software image type. For example, the Cisco Catalyst 9115AX series AP and the Cisco Catalyst 9120AX series AP use the ap1g7 AP software image file. So, the Cisco Embedded Wireless Controller network in this example containing Cisco Catalyst 9115AX series and 9120AX series APs is a homogenous network.

The embedded wireless controller CLI can only be used to set the mode for image download as desktop (HTTP). The Cisco Embedded Wireless Controller GUI has to be used to configure and trigger network software upgrade (pre-download) using the desktop (HTTP) image download method.

Prerequisites for Image Download

- Connectivity to an external (TFTP or SFTP) server is required for image download during AP join in a Cisco Embedded Wireless Controller network.
- Connectivity to a PC or laptop is required for image download during network software upgrade in a Cisco Embedded Wireless Controller network.
- All APs should be connected to the embedded wireless controller for image download in the network software upgrade (pre-download) scenario.
- For image upgrade, you must not configure a preferred-master. If you configure a preferred-master, ensure that it points to the currently active AP, which is displayed in the **show wireless ewc-ap redundancy summary** command.

  If a different AP is configured as the preferred-master, the upgrade process will not take place in the **install activate** step. If the upgrade does not take place, you should either remove the preferred-master

configuration, or re-configure the preferred-master to match the AP that is currently active, and then run the **install activate** command, again.

•

Configuring Image Download Profile

You need to configure the image download profile for both the AP join image download and pre-download scenarios. The only profile supported is *default*. In a Cisco Embedded Wireless Controller network, only one site tag is supported, the *default-site-tag*. The *default* image download profile is attached to the *default-site-tag*.

Configuring TFTP Image Download (GUI)

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as TFTP.

Step 3 In the **Image Server** field, enter the TFTP server IP address.

Step 4 In the **Image Path** field, enter the absolute or relative path to the software image bundle.

Step 5 Choose one of the following:

- **Save:** Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
- **Save & Download:** Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
- **Activate:** Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
- **Cancel:** Choose this option to cancel any changes made to the image download profile.

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.

Option	Description
Cancel	Choose this option to cancel any changes made to the image download profile.

Configuring TFTP Image Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device (config)# wireless profile image-download default`	Configures the default AP profile.
Step 3	**image-download-mode tftp** **Example:** `Device(config-wireless-image-download-profile)# image-download-mode tftp`	Configure image download using TFTP.
Step 4	**tftp-image-server** *server-ip* **Example:** `Device(config-wireless-image-download-profile-tftp)# tftp-image-server 10.1.1.1`	Configure the TFTP server for image download by specifying the IPv4 or IPv6 *server-ip* address.
Step 5	**tftp-image-path** *server-path* **Example:** `Device(config-wireless-image-download-profile-tftp)# tftp-image-path /download/object/stream/images/ap-images`	Configure the absolute or relative path to the software image on the TFTP server.
Step 6	**end** **Example:** `Device(config-wireless-image-download-profile-tftp)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring SFTP Image Download (GUI)

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as SFTP.

The SFTP port is not configurable and is fixed at 22.

Step 3 In the **Image Server** field, enter the SFTP server IP address.

Step 4 In the **Image Path** field, enter the path to the software image bundle.

Step 5 In the **User Name** field, enter the SFTP server username.

Step 6 Choose the appropriate **Password Type** from Unencrypted or AES Encrypted.

Step 7 In the **Password** field, enter the SFTP server password.

Step 8 Choose one of the following:

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Configuring SFTP Image Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile image-download default** **Example:**	Configures the default AP profile.

	Command or Action	Purpose
	`Device (config)# wireless profile image-download default`	
Step 3	**image-download-mode sftp** **Example:** `Device(config-wireless-image-download-profile)# image-download-mode sftp`	Configure image download using SFTP.
Step 4	**sftp-image-server** *server-ip* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-image-server 10.1.1.1`	Configure the SFTP server for image download by specifying the IPv4 or IPv6 *server-ip* address.
Step 5	**sftp-image-path** *server-path* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-image-path /download/object/stream/images/ap-images`	Configure the path to the software image on the SFTP server.
Step 6	**sftp-username** *username* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-username test`	Specify the username to log in to the SFTP server for image download.
Step 7	**sftp-password** {**0** | **8**} *password* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-password 0 password1`	Specify the password associated with the above username to download the image from the SFTP server. You need to re-enter the password to confirm the entry. To configure an AES encrypted password, specify `8`, else specify `0` to configure an unencrypted password.
Step 8	**end** **Example:** `Device(config-wireless-image-download-profile-tftp)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Desktop (HTTP) Image Download (GUI)

- Image download using desktop (HTTP) is only enabled in a homogeneous network, that is a network containing APs that have the same image type.
- Image download using desktop (HTTP) can only be configured from the GUI.
- The CLI can only be used to set the image download mode to desktop (HTTP).

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as Desktop (HTTP).

Step 3 In the **Controller Image** field, navigate to the embedded wireless controller software image on your computer or laptop desktop.

Step 4 In the **AP Image** field, navigate to the AP software image on your computer or laptop desktop.

The GUI displays the name of the AP image to be used. Depending on the AP model, the name of the AP image varies.

Step 5 Choose one of the following:

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Initiating Pre-Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**clear ap predownload statistics**	Clear AP predownload statistics.
Step 2	**install remove profile default**	Remove the image download profile. Choose **Y** to remove the profile or choose **N** to cancel.
Step 3	**install add profile default**	Download the controller and AP software image from the embedded wireless controller. The controller image is sent to all Cisco Embedded Wireless Controller-capable APs.

	Command or Action	Purpose
		The AP image is downloaded to all APs sharing the same image type.
Step 4	**show wireless ewc-ap predownload status**	Monitor the overall software download status. The download is successful when the status message is `Controller Image Predownload to EWC Capable APs Complete`.
Step 5	**install activate**	Activate the network after upgrade. All the subordinate APs get the new AP image and reboot. Once all APs are rebooted, the embedded wireless controller also reboots. **Note** The network can also be activated if the controller image is downloaded but all APs have not received the AP image via predownload. **Important** If the network is activated during partial predownload success, and a Cisco Embedded Wireless Controller-capable AP with old controller software becomes the controller, then the network will not get upgraded to the new image.
Step 6	**show install summary**	Verify the current image status after rebooting. If the status is `Activated and Uncommitted`, proceed to Step 7, else wait.
Step 7	**install commit**	Commits the current software image once the embedded wireless controller comes up after rebooting.

Verifying Image Download

To monitor the overall progress of the software download process during predownload, run the following command.

```
Device# show wireless ewc-ap predownload status
```

The following are the various status messages indicating the status of the predownload operation. These are displayed when you run the **show wireless ewc-ap predownload status** command:

- None
- Controller Image Download Initiated

- Controller Image Download In Progress
- Controller Image Download Complete
- Controller Image Download Failed
- AP Image Predownload Initiated
- AP Image Predownload In Progress
- AP Image Predownload Complete
- AP Image Predownload Unsupported
- AP Image Predownload Failed
- Controller Image Predownload to EWC Capable APs In Progress
- Controller Image Predownload to EWC Capable APs Complete
- Controller Image Predownload to EWC Capable APs Failed
- Image Activation Succeeded
- Image Activation Failed
- Invalid State

To view the AP image predownload statistics, run the following command:

```
Device# show wireless ewc-ap ap image predownload status
Total number of APs                          : 5
Total number of EWC capable APs              : 4
Number of APs
        Initiated                            : 0
        Predownloading AP image              : 0
        Predownloading Controller image      : 1
        Completed predownloading AP          : 5
        Completed predownloading Controller : 0
        Failed to Predownload AP             : 0
        Failed to Predownload Controller     : 0
AP Name                Primary Image (AP/Controller)          Backup Image (AP/Controller)
          Predownload Status          Predownload Version                    AP Image
Role    Retries  AP image       Controller image

                                                                                Type
               ETA/Percent    ETA/Percent
-----------------------------------------------------------------------------------------------------------
APXXXX.9XXX.8FXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.2.02.0.XXXX
         Complete                  17.2.2.2       /17.2.02.0.2XXX         ap1g7       Slave
   0        00:00:00/100%  00:00:00/  0%
APXXXX.5XXX.71XX       17.3.0.85      /                       17.2.2.2       /
           Complete                   17.2.2.2       /                       ap1g5
Master  0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.59XX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.2.02.0.XXXX
         Complete                  17.2.2.2       /                       ap1g7       Slave
   0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.5AXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.3.01.0.XXX
          Controller Predownloading  17.2.2.2       /                       ap1g7
Master  0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.5BXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /
           Complete                   17.2.2.2       /                       ap1g7
Slave   0        00:00:00/100%  00:00:00/  0%
```

To view details of the AP acting as the primary image , use the following command:

```
Device# show wireless ewc-ap image-master
Image Master List
Image Name: ap1g7
-------------------------------------------------------------------------------------------------------------
Master AP MAC                AP                          AP                 Controller
         Controller
                Predownload In Progress   Predownload Complete     Predownload In Progress
  Predownload Complete
-------------------------------------------------------------------------------------------------------------
c0XX.eXXX.90XX    No                          No                       No
   Yes
Image Name: ap1g5
-------------------------------------------------------------------------------------------------------------
Master AP MAC                AP                          AP                 Controller
         Controller
                Predownload In Progress   Predownload Complete     Predownload In Progress
  Predownload Complete
-------------------------------------------------------------------------------------------------------------
70XX.1XXX.4bXX    No                          No                       No
   Yes
```

To check the image download status on all the APs, run the following command:

```
Device# show ap image
```

To check AP status during image download, run the following command:

```
Device# show ap summary
```

To monitor efficient AP join status, run the following command:

```
Device# show ap master list
```

To view the details of the last AP image download attempt, run the following command:

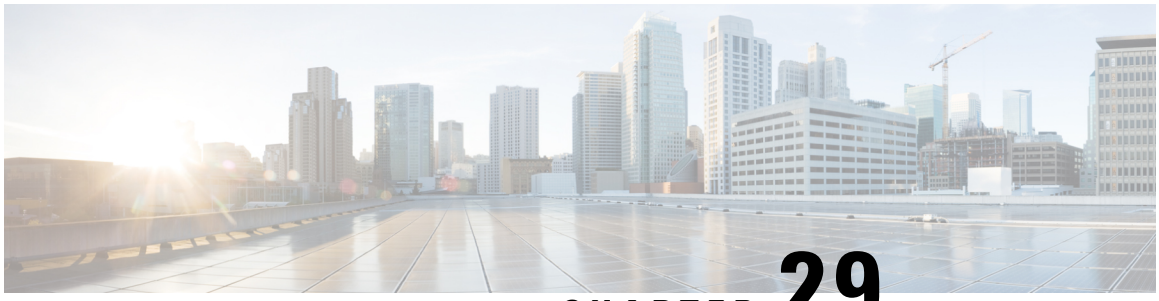
```
Device# show wireless stats ap image-download
```

To check the current status of the upgraded image, run the following command:

```
Device# show install summary
```

To check the download status from external servers (TFTP or SFTP), run the following command:

```
Device# show install log
```



CHAPTER 29

Conditional Debug, Radioactive Tracing, and Packet Tracing

- Conditional debugging, on page 253
- Packet tracing, on page 259

Conditional debugging

A conditional debugging is a diagnostic feature that

- enables selective activation of debugging and logging for specific features based on conditions you define
- allows you to define custom conditions such as interface, IP address, or MAC address to narrow debug scope, and
- helps reduce system resource usage and performance impact in large, complex networks.

Use this feature in systems with many supported features.

Conditional debug allows granular debugging in networks operating at large scale with many features. It allows you to observe detailed debug output for specific instances within the system.

In contrast, the general debug command produces output without filtering feature objects being processed. The general debug command uses significant system resources and affects performance.

Radioactive tracing

A radioactive tracing is a diagnostic tracing mechanism that

- stitches together a chain of execution for operations of interest across the system, at an increased verbosity level
- enables conditional printing of detailed debug information across threads, processes and function calls, and
- troubleshoot system features, especially First-Hop Security (FHS) and mesh networking functions.

Note

-
- The radioactive tracing filter does not work, if the certificate is not valid.
- To debug mesh features effectively, add both Ethernet and Radio MAC addresses as conditional MACs for RA tracing when you collect logs.
- To enable debugging for wireless IPs, use the **debug platform condition feature wireless ip** *ip-address* command.

Feature history for conditional debug and radioactive tracing

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 13: Feature history for conditional debug and radioactive tracing

Feature Name	Release Information	Feature Description
Conditional debug and radioactive tracing	Cisco IOS XE 17.11.1	Conditional debugging is a diagnostic feature that selectively activates debugging and logging based on defined conditions to optimize resource usage and performance in complex networks, while radioactive tracing is a diagnostic mechanism that stitches together execution chains with increased verbosity, enabling conditional detailed debug information across system components for troubleshooting specific features.

Conditional debugging and radioactive tracing

A conditional debugging and radioactive tracing feature is a diagnostic mechanism that

- combines the precision of conditional debugging to target specific execution contexts
- leverages radioactive tracing to follow code paths across multiple control flows, and
- enables use of a single debug CLI to debug all execution contexts related to the condition.

Note Use the **clear platform condition all** command to remove the debug conditions applied to the platform.

Location of tracefiles

Tracefiles are generated for each process and saved in either the `/tmp/rp/trace` or `/tmp/fp/trace` directory. Each trace log is written to a file of 1 MB. The directory can hold up to a maximum of 25 such files

for a given process. When a tracefile in the `/tmp` directory reaches its 1MB limit or the configured size, it is rotated out to an archive location in the `/crashinfo` partition under the `tracelogs` directory.

- Process-name_Process-ID_running-counter.timestamp.gz

  Example: IOSRP_R0-0.bin_0.14239.20151101234827.gz

- Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz

  Example: wncmgrd_R0-0.27958_1.20180902081532.bin.gz

The `/tmp` directory holds only a single tracefile for a given process. Once a tracefile reaches its size limit, the system moves it to `/crashinfo/tracelogs`. The archive directory stores up to 25 files and replaces the oldest file with each new file from `/tmp`. File size varies by process. Some processes use larger files that can reach 10 MB.

- The number of files stored in the `tracelogs` directory also depends on the process. For example, the WNCD process may use a limit of up to 400 files per instance, depending on the platform.
- You can verify these logs (per-process) using the **show platform software trace message** *process_name* **chassis active R0** command.

Configure conditional debugging (GUI)

Enable targeted troubleshooting by configuring conditional debugging through the GUI, allowing you to capture and analyze logs for specific MAC or IP addresses.

Procedure

Step 1 Choose **Troubleshooting** > **Radioactive Trace**. Click **Add**.

Step 2 Enter the **MAC/IP Address**.

Step 3 Click **Apply to Device**.

Step 4 Click **Start** to start or **Stop** to stop the conditional debug.

Step 5 Click **Generate** to create a radioactive trace log.

Step 6 Click the radio button to set the time interval.

Step 7 Click the **Download Logs** icon that is displayed next to the trace file name, to download the logs to your local folder.

Step 8 Click the **View Logs** icon that is displayed next to the trace file name, to view the log files on the GUI page.

Step 9 Click **Load More** to view more lines of the log file. Click **Apply to Device**.

Configure conditional debugging (CLI)

Enable targeted troubleshooting for wireless features by filtering debug output to specific APs, clients, or IP addresses using commands.

Procedure

Step 1 Configure conditional debugging for a feature using the specified MAC address.

Example:

```
Device# debug platform condition feature wireless mac b838.61a1.5433
```

Note
This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

Step 2 Start conditional debugging.

Example:

```
Device# debug platform condition start
```

This starts radioactive tracing if there is a match on one of the conditions above.

Note
This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

Step 3 Display the current conditions set.

Example:

```
Device# show platform condition
```

```
Device# show debugging
```

Step 4 Stop conditional debugging.

Example:

```
Device# debug platform condition stop
```

This will stop radioactive tracing.

Note
This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

Step 5 Display the logs from the latest wireless profile.

Example:

```
Device# show logging profile wireless [ counter start last {20 minutes} | filter mac
{mac-address}  to-file bootflash:logs.txt]
```

Note
You can use either the **show logging profile wireless** command or **show logging process** command to collect the logs.

Step 6 Display the logs collection specific to the process.

Example:

```
Device# show logging process wncd to-file flash:wncd.txt
```

Step 7 Clear all conditions.

Example:

```
Device# clear platform condition all
```

What to do next

The command **request platform software trace filter-binary wireless** *mac-address* generates 3 flash files:

- collated_log_<.date..>
- mac_log <..date..>
- mac_database .. file

The *mac_log <..date..>* file is the most important because it contains messages related to the MAC address under investigation. The **show platform software trace filter-binary** command generates the same flash files and displays the mac_log on the screen.

Recommended workflow for trace files

The recommended workflow for trace files is listed below:

- To request the tracelogs for a specific time period, such as 1 day, use the command: Device# **show logging process wncd to-file flash:wncd.txt**.
- The system generates a text file of the tracelogs in the location /flash:
- Copy the file off the device so the tracelogs can be used to work offline. For more details on copying files, see section below.
- Delete the tracelog file (.txt) from the /flash: location to ensure enough space on the device for other operations.

Copy tracefiles off the box

This is an example of the tracefile:

```
Device# dir flash:/tracelogs
Directory of crashinfo:/tracelogs/
50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
50603 -rwx 991 Sep 22 2015 11:12:08 +00:00 fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00 reflector_803_R0-0.bin_0.1312.20150922111116.gz

               50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00
IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
```

The trace files can be copied using one of the various options shown below:

```
Device# copy flash:/tracelogs ?
  crashinfo: Copy to crashinfo: file system
  flash: Copy to flash: file system
```

```
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system
```

The general syntax for copying onto a TFTP server is as follows:

```
Device# copy source: tftp:
Device# copy crashinfo:/tracelogs/IOSRP_R0-0.bin_0.14239.20151101234827.gz tftp:
Address or name of remote host []? 2.2.2.2
Destination filename [IOSRP_R0-0.bin_0.14239.20151101234827.gz]?
```



Note Clear the generated report and archive files from the switch to ensure that sufficient flash space remains available for tracelog and other uses.

Configuration examples for conditional debugging

This is an output example of the **show platform condition** command.

```
Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
------------------------------------------------------------------------------------------|---------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
```

This is an output example of the **show debug** command.

```
Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start
Conditions Direction
------------------------------------------------------------------------------------------|---------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
Packet Infra debugs:
Ip Address Port
------------------------------------------------------|----------
```

Verify conditional debugging

The table shown below lists the various commands that can be used to verify conditional debugging:

Command	Purpose
show platform condition	Displays the current conditions set.
show debug	Displays the current debug conditions set.
show platform software trace filter-binary	Displays logs merged from the latest tracefile.
request platform software trace filter-binary	Displays historical logs of merged tracefiles on the system.

Example: verify radioactive tracing log for SISF

This is an output example of the **show platform software trace message ios chassis active R0 | inc sisf** command.

```
Device# show platform software trace message ios chassis active R0 | inc sisf
2017/10/26 13:46:22.104 {IOSRP_R0-0}{1}: [parser]: [5437]:  UUID: 0, ra: 0 (note):  CMD:
'show platform software trace message ios switch active R0 | inc sisf' 13:46:22 UTC Thu Oct
 26 2017
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Unlocking, count is now 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Unlocking, count is now 1
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Setting State to 2
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Start timer 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc  Timer value/granularity for 0 :299998/1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Updated Mac Timer : 299998
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Before Timer :  350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Timer 0, default value is 350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   Allocating timer wheel for 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc No timer running
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Granularity for timer MAC_T1  is 1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Current State :MAC-STALE, Req Timer : MAC_T1  Current Timer
 MAC_T1
```

Packet tracing

A packet tracing is a network diagnostic tool that

- tracks the flow and processing of individual data packets through a network device

- provides detailed per-packet information based on user-defined conditions for troubleshooting purposes, and
- helps identify network issues such as misconfiguration, capacity overload, and software bugs.

The packet tracing feature explains how to perform data plane packet tracing for Cisco Catalyst 9800 Series Wireless Controller for Cloud software.

To trace packets on the controller:

1. Enable conditional debugging on selected packets or traffic to trace.
2. Enable packet tracing (per-AP or per-Client).



Note Use per-AP conditional debugging with a MAC address filter when the AP and controllers are in the same VLAN. If they are in different VLANs, per-AP packet tracing with a MAC address does not capture packets because the MAC address varies.

Limitation of Conditional Debugging Packet Tracing

The MAC or IP filter applies only to the outer Ethernet or IP header. If a packet is CAPWAP encapsulated, the MAC or IP filter does not affect the inner 802.11 MAC or IP header.

Configure conditional debugging packet tracing (CLI)

Enable and manage in-depth packet tracing for troubleshooting and diagnostics while minimizing resource usage using commands.

Procedure

Step 1 Enable the privileged **EXEC** mode.

Example:

```
Device# enable
```

Enter your password, if prompted.

Step 2 Configure packet tracing to capture the last set of packets.

Example:

```
Device# debug platform packet-trace packet 8192 circular fia-trace data-size 2048
```

packet-count: Here, the valid range is from 16 to 8192.

data-size: Here, the valid range is from 2048 to 16384 bytes.

Step 3 Configure packet tracing for a copy of packet data.

Example:

```
Device# debug platform packet-trace copy packet both size 2048
```

packet-size: Here, the valid range is from 16 to 2048 bytes.

Step 4 Enable conditional debugging for an interface, MAC, or IP filter.

Example:

Enable conditional debugging for TenGigabitEthernet 0/0/0 and match packets whose source and destination MAC is 192.0.2.1:

```
Device# debug platform condition interface {interface-name TenGigabitEthernet | cpp} 0/0/0
 {mac | ipv4 | match} 192.0.2.1 {both | ingress | egress}
```

An interface refers to any physical port, port channel, internal vlan, SVI, or wireless client.

Step 5 Start conditional debugging packet tracing.

Example:

```
Device# debug platform condition start
```

Step 6 Stop conditional debugging packet tracing.

Example:

```
Device# debug platform condition stop
```

Step 7 Redirect all traced packets to bootflash.

Example:

```
Device# show platform hardware chassis active qfp feature packet-trace packet all | redirect
 bootflash:packet_trace.txt
```

Convert the packet_trace.txt to pcap and download the pcap files. You can do so using this link:

http://wwwin-dharton-dev.cisco.com/pactrac2pcap.html

Configure conditional debugging packet tracing per AP (CLI)

Enable conditional debugging to trace CAPWAP packets for a specific AP by filtering with MAC or IP address using commands.

Procedure

Step 1 Enable privileged **EXEC** mode.

Example:

```
Device# enable
```

Enter your password, if prompted.

Step 2 Enable conditional debugging with MAC filter.

Example:

```
Device# debug platform condition interface {interface-name TenGigabitEthernet | cpp} 0/0/0
 {mac 192.0.2.1 | access-list acl-name | ipv4 | match} {both | ingress | egress}
```

Here, the CLI matches the packets whose source or destination MAC address is 192.0.2.1

Step 3 Enable conditional debugging with inline MAC ACL.

Example:

```
Device# debug platform condition interface TenGigabitEthernet 0/0/0 match mac { H.H.H. |
any | host} 192.0.2.1 {both | ingress | egress}
```

Step 4 Enable conditional debugging with IP filter.

Example:

```
Device# debug platform condition interface TenGigabitEthernet 0/0/0 ipv4 {192.0.2.1 |
access-list ip-acl-name| both | egress | ingress} {both | egress | ingress}
```

0/0/0: is the GigabitEthernet interface number. The valid range is from 1 to 32.

Configure conditional debugging packet tracing per client (GUI)

Enable targeted packet tracing for individual clients to facilitate in-depth troubleshooting and analysis using GUI.

Procedure

Step 1 Choose **Troubleshooting** > **Radioactive Trace**.

Step 2 Click **Add**.

Step 3 In the Add MAC/IP Address window, enter the **MAC/IP Address**.

Step 4 Click **Apply to Device**.

Configure conditional debugging packet tracing per client (CLI)

Enable conditional debugging to trace packets for a specific wireless client, helping to diagnose network issues more efficiently using commands.

Procedure

Step 1 Enable privileged **EXEC** mode.

Example:

```
Device# enable
```

Enter your password, if prompted.

Step 2 Enable conditional debugging for a wireless client interface.

Example:

```
Device# debug platform condition interface {intf-name | cpp 0xa0000001} { mac | ipv4 | match
 [ipv4 | ipv6 | mac]} protocol icmp host 192.0.2.1 host 192.0.2.1 {both | ingress | egress}
```

cpp-handle-index: The valid range is from 1 to 4294967295.

Verify conditional debugging packet tracing configuration

To view the summary of the traced packet, use this command:

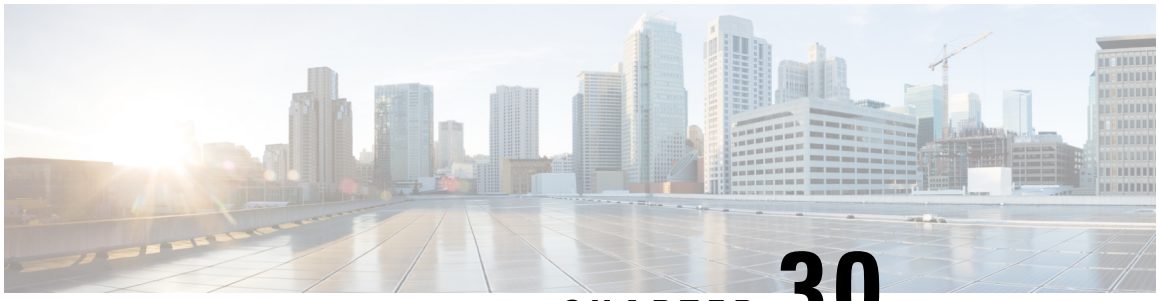
```
Device# show platform packet-trace summary
```

To view a specific traced packet, use this command:

```
Device# show platform packet-trace packet <packet number>
```

To view the wireless client interface handle, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
mac-address client-mac details
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
mac-address 8825.93b0.b51f details
Client Details for client cpp_if_handle: 0x34
Name : WLCLIENT-IF-0x00a0000001
Mac Addr : 8825.93b0.b51f
pal_if_handle : 0xa0000001
Mobility State : LOCAL
Multicast Action : FORWARD
Auth State : RUN
```



CHAPTER 30

Aggressive Client Load Balancing

- Aggressive client load balancing, on page 265

Aggressive client load balancing

Aggressive client load balancing is a wireless network feature that

- distributes wireless clients across lightweight APs managed by the same controller
- redirects client association attempts from overutilized APs to less busy APs using standardized response codes, and
- helps optimize overall network performance by preventing AP overload.

Guidelines of aggressive client load balancing

When a wireless client attempts to associate to a lightweight AP, the associated response packets are sent to a client with an 802.11 response packet including status code 17. This code 17 indicates that the corresponding AP is busy. The AP does not respond with 'success' if the AP threshold is not met. If the AP utilization threshold is exceeded and another less busy AP hears the client request, the AP responds with code 17 (AP busy).

If AP1 has more clients than AP2 during the load-balancing window, AP1 is considered busier. When a client attempts to associate to AP1, the client receives an 802.11 response packet with status code 17, indicating that the AP is busy, and the client attempts to associate to a different AP.

Configure the embedded wireless controller to deny client associations up to 10 times. If a client tries to connect 11 times, it will connect on the 11th attempt. Enable or disable load balancing on a particular WLAN. This option is helpful if you need to exclude time-sensitive voice clients from load balancing.

Note For a FlexConnect AP, the association is locally handled. The controller makes load-balancing decisions. A FlexConnect AP sends an initial response to the client before knowing the result of the calculations in the controller. Load balancing does not work when the FlexConnect AP is in standalone mode.

In this mode, the FlexConnect AP does not send a (re)association response with status 17 for load balancing. Instead, it first sends a (re)association response with status zero (success), and then sends a deauthentication with reason five.

Restrictions of aggressive client load balancing

- This feature is not supported on the APs joined on default-site-tag.
- This feature is not supported on the APs across different named site-tags.
- This feature is supported only on the APs within a named-site-tag.

Enable aggressive client load balancing (GUI)

Ensure clients are evenly distributed across APs to maximize WLAN performance using GUI.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.

Step 2 Select a **WLAN** to view the **Edit WLAN** window.

Step 3 Click on the **Advanced** tab.

Step 4 Check the **Load Balance** check box to enable the feature.

Step 5 Click **Update & Apply to Device**.

Configure aggressive client load balancing (GUI)

Set the number of clients and denial count for aggressive load balancing to optimize wireless network performance using GUI.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Advanced**.

The **Load Balancing** window appears.

Step 2 In the **Aggressive Load Balancing Window (clients)** field, enter the number of clients for the aggressive load balancing client window.

Step 3 In the **Aggressive Load Balancing Denial Count** field, enter the load balancing denial count.

Step 4 Click **Apply**.

Configure aggressive client load balancing (CLI)

Configure aggressive client load balancing to optimize WLAN performance and distribute clients across APs using commands.

Procedure

Step 1 Enter the global configuration mode and specify the WLAN name.

Example:

```
Device# configure terminal
Device(config)# wlan test-wlan
```

Step 2 Disable the WLAN.

Example:

```
Device(config-wlan)# shutdown
```

Step 3 Configure a guest embedded wireless controller as mobility controller, in order to enable client load balance to a particular WLAN.

Example:

```
Device(config-wlan)# load-balance
```

Configure the WLAN security settings as the WLAN requirements.

Step 4 Enable WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 5 Return to the privileged EXEC mode and enter the global configuration mode.

Example:

```
Device(config)# end
Device(config)# configure terminal
```

Step 6 Configure the load balancing denial count.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} load-balancing denial 10
```

Step 7 Configure the number of clients for the aggressive load balancing client window.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} load-balancing window 10
```

Step 8 Return to the privileged EXEC mode.

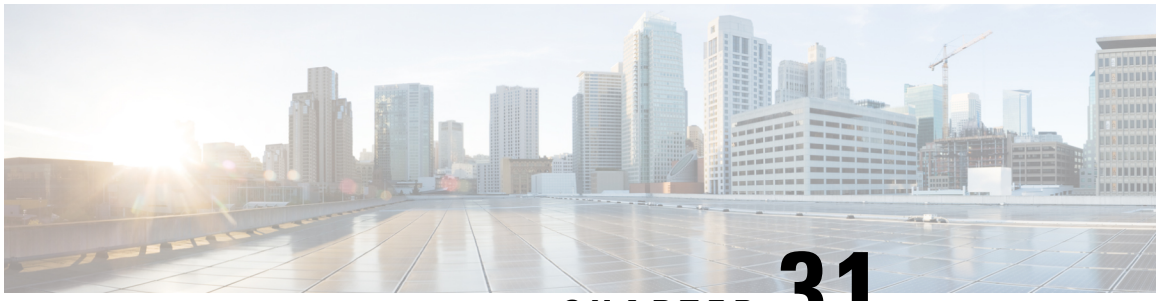
Example:

```
Device(config-wlan)# end
```

Step 9 Display a filtered section of the current configuration.

Example:

```
Device# show running-config | section test-wlan
```



CHAPTER 31

Accounting Identity List

- Configure accounting identity list (GUI), on page 269

Configure accounting identity list (GUI)

Create a method list for accounting identity using the GUI to control and monitor user access to your network.

Procedure

Step 1	Choose **Configuration** > **Security** > **AAA**.
Step 2	In the **AAA Method List** tab, go to the **Accounting** section, and click **Add**.
Step 3	In the **Quick Setup: AAA Accounting** window that is displayed, enter a name for your method list.
Step 4	Select the type of authentication as identity, in the **Type** drop-down list.
Step 5	Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click > icon to move them to the **Assigned Server Groups** list.
Step 6	Click **Save & Apply to Device**.

Configure accounting identity list (CLI)

Enable RADIUS accounting with an identity list to log user actions and network usage events using commands.

When a user successfully executes an action, the RADIUS accounting server logs the changed attributes, user ID, remote host, date and time of execution, authorization level, description of the action performed, and values provided.

Follow the procedure given below to configure accounting identity list.

Before you begin

Configure the RADIUS server and AAA group server.

Procedure

Enable accounting to send a start-record accounting notice when a client is authorized and a stop-record at the end.

Example:

```
Device(config)# aaa accounting identity user1 start-stop group aaa-test
```

Note
You can also use the default list, instead of a named list.

Configure client accounting (GUI)

Enable client accounting on a specified policy profile through the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click the **Policy Profile Name**. In the **Edit Policy Profile** window, go to the **Advanced** tab.

Step 3 From the **Accounting List** drop-down, select the appropriate accounting list for this policy profile. The policy profile uses the accounting type you selected before gaining access to the network.

Step 4 Click **Save & Apply to Device**.

Configure client accounting (CLI)

Enable client accounting on a WLAN policy profile to track user activity and support auditing with an accounting server using commands.

Procedure

Step 1 Configure WLAN policy profile and enter the wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy default-policy-profile
```

Step 2 Disable the policy profile.

Example:

```
Device(config-wireless-policy)# shutdown
```

Step 3 Set the accounting list.

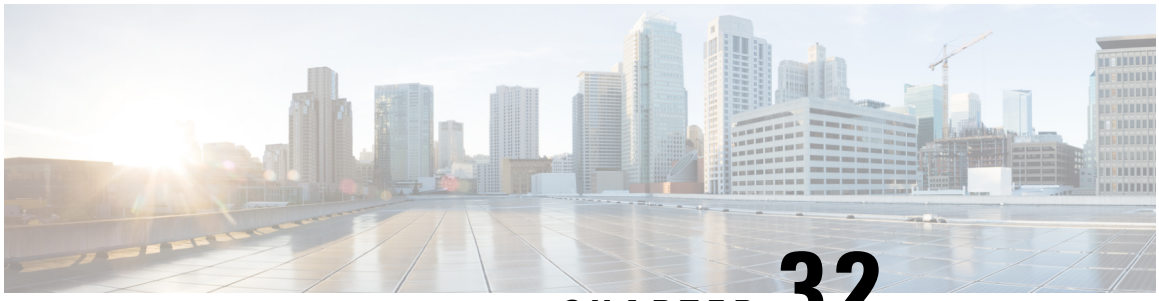
Example:

```
Device(config-wireless-policy)# accounting-list user1
```

Step 4 Enable the policy profile.

Example:

```
Device(config-wireless-policy)# no shutdown
```



CHAPTER 32

Volume metering

A volume metering feature is an access point data reporting capability that

- allows you to set the interval for client accounting statistics updates from the access point (AP) to the network controller,
- supports configurable reporting frequencies from five to 90 seconds, and
- helps minimize delays in updating usage data for connected devices.

Additional information

By default, the access point sends client accounting updates to the controller every 90 seconds, which then forwards this information to the RADIUS server. With volume metering features, you can adjust the reporting interval from five seconds to 90 seconds.

This flexibility enables network administrators to better control the timeliness of usage data, ensuring more immediate reporting and improved network accounting accuracy.

Example

If you want to decrease the delay in receiving accounting data for a device, you can configure the reporting interval to 10 seconds so that the access point sends updates to the controller—and subsequently to the RADIUS server—much more frequently than the default 90 seconds.

This supports more accurate, real-time accounting of network usage.

• Configure volume metering (CLI), on page 273

Configure volume metering (CLI)

Configure device parameters to control how often the system reports wireless usage data and sends accounting updates to the Remote Authentication Dial-In User Service (RADIUS) server.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an access point (AP) profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile profile-name
```

Step 3 Configure the 2.4 GHz (802.11ax) reporting interval.

Example:

```
Device(config-ap-profile)# dot11 24ghz reporting-interval reporting-interval
```

Step 4 Configure the 5 GHz (802.11ax) reporting interval.

Example:

```
Device(config-ap-profile)# dot11 5ghz reporting-interval reporting-interval
```

Step 5 Return to global configuration mode.

Example:

```
Device(config-ap-profile)# exit
```

Step 6 Set the time interval (in minutes) at which the embedded wireless controller sends interim accounting updates for the client to the RADIUS server.

Example:

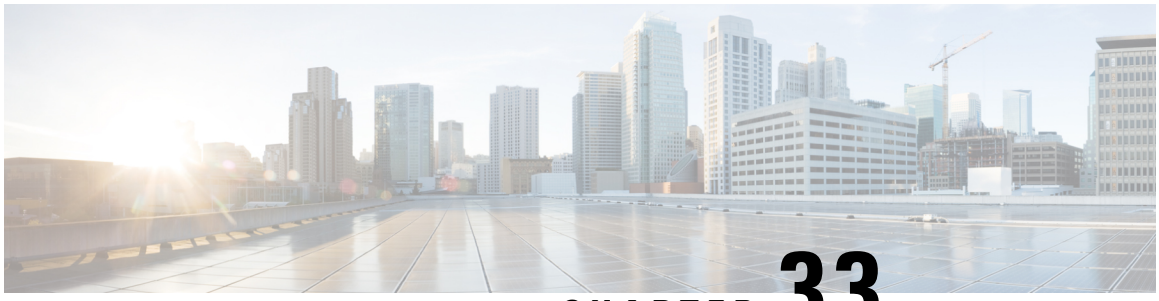
```
Device(config)# aaa accounting update periodic interval-in-minutes
```

Step 7 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config)# exit
```

The device reports volume metering data at your specified intervals and sends accounting updates to the RADIUS server.



CHAPTER 33

Enabling Syslog Messages in Access Points and Controller for Syslog Server

- Syslog messages in access points and controllers, on page 275
- AP logging to the in-memory buffer and flash, on page 278

Syslog messages in access points and controllers

A syslog message in access points and controllers is a log communication that

- captures and records operational events generated by each device
- is routed to various destinations, including memory buffers, terminal sessions, device storage, or external syslog servers, and
- allows administrators to configure logging independently on access points and controllers to meet specific network requirements.

Configuring message logging in the IOS XE controller

System Message Logging in Cisco Catalyst 9800 Series Controllers is a platform-independent IOS and IOS XE feature. For more details, see:

- System Message Logging
- Configuration Logger Persistency chapter in *System Management Configuration Guide*
- Logging to Local Nonvolatile Storage chapter in *System Management Configuration Guide*
- Embedded Syslog Manager (ESM) chapter in *System Management Configuration Guide*
- Configuration Change Notification and Logging chapter in *System Management Configuration Guide*

Configure the syslog server for the controller (GUI)

Enable centralized log management by forwarding controller system logs to a designated syslog server through the controller GUI.

Procedure

Step 1 Choose **Troubleshooting** > **Logs**.

Step 2 Click **Manage Syslog Servers**.

Step 3 In **Log Level Settings**, from the **Syslog** drop-down list, choose a severity level.

Step 4 From the **Message Console** drop-down list, choose a logging level.

Step 5 In **Message Buffer Configuration**, from the **Level** drop-down list, choose a server logging level.

Step 6 In **IP Configuration** settings, click **Add**.

Step 7 From the **Server Type** drop-down list, choose **IPv4**, **IPv6**, or **FQDN**.

Step 8 If you choose **IPv4** or **IPv6**, enter the **Server Address**. If you choose **FQDN**, enter the **Host Name**. Then choose the IP type and the appropriate **VRF Name** from the drop-down lists.

To delete a syslog server, click **x** next to the appropriate server entry under the **Remove** column.

Note
Spaces are not allowed in the host name.

Step 9 Click **Apply to Device**.

Note
When you click **Apply to Device**, the system applies the changes. If you click **Cancel**, the system discards the changes.

After you apply the configuration, log messages are forwarded to the specified syslog server.

Configure a syslog server for a controller (CLI)

Configure the controller to send system and event messages to a remote syslog server for monitoring.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter the syslog server IP address and configure its parameters.

Example:

```
Device(config)# logging hostname ipv6
```

Step 3 Enable the facility parameter for syslog messages.

Example:

```
Device(config)# logging facility {auth | cron | daemon | kern | local0 | local1 | local2 |
 local3 | local4 | local5 | local6 | local7 | lpr | mail | news | sys10 | sys11 | sys12 |
sys13 | sys14 | sys9 | syslog | user | uucp}
```

Example:

```
Device(config)# logging facility syslog
```

You can enable these facility parameters for syslog messages:

- **auth** Authorization system.
- **cron** Cron facility.
- **daemon** System daemons.
- **kern** Kernel.
- **local0** to **local7** Local use.
- **lpr** Line printer system.
- **mail** Mail system.
- **news** USENET news.
- **sys10** to **sys14** and **sys9** System use.
- **syslog** Syslog itself.
- **user** User process.
- **uucp** Unix-to-Unix copy system.

Step 4 Set the logging level for the syslog server.

Example:

```
Device(config)# logging trap {severity-level | alerts | critical | debugging | emergencies
 | errors | informational | notifications | warnings}
```

Example:

```
Device(config)# logging trap 2
```

severity-level Refers to the logging severity level. The valid range is from zero to seven.

These are the syslog server logging levels:

- **emergencies** Signifies severity 0. The system is not usable.
- **alerts** Signifies severity 1. Immediate action is required.
- **critical** Signifies severity 2. Critical conditions.
- **errors** Signifies severity 3. Error conditions.
- **warnings** Signifies severity 4. Warning conditions.
- **notifications** Signifies severity 5. Normal but significant conditions.
- **informational** Signifies severity 6. Informational messages.
- **debugging** Signifies severity 7. Debugging messages.

Note
Select a syslog level to view the supported levels. Selecting a level also enables all lower levels.

If you enable the *critical* syslog level, the system also enables lower levels—*alerts* and *emergencies*.

Step 5 Enter privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, press **Ctrl-Z** to exit global configuration mode.

The controller now forwards syslog messages to the specified syslog server as configured.

AP logging to the in-memory buffer and flash

AP always log messages to an in-memory buffer. Once the buffer reaches 40 KB, its contents are automatically written to flash memory, and a new buffer is created.

Administrators can manage and view these logs using AP commands.

- **show logging** command to display the contents of the in-memory logging buffer
- **show flash syslogs** command to list all log files stored in flash, along with other diagnostic files
- **<filename>** command to display the contents of an individual log file stored in flash
- **copy syslogs <filename>** command to transfer a specific syslog file to an external server. To see available options for this command, use **copy syslogs <filename>**

AP logging to terminal

Access points support real-time logging of messages to an active SSH terminal session, which can be enabled or disabled using the **terminal monitor** and **terminal monitor disable** commands.

Administrators can enable real-time logging to an SSH terminal session using the **terminal monitor** command. To disable real-time logging to the session, use the **terminal monitor disable** command. In addition to SSH terminal sessions, APs send a subset of log messages to the serial console, providing another method for real-time monitoring.

Configure AP logging to a syslog server

Use the **syslog** command in the AP join profile to configure the destination IP address for syslog messages and manage which messages are sent based on severity and facility levels.

Configure the syslog host

- Use the **syslog host <IP address>** command to specify the destination IP address for syslog messages.
- By default, the syslog host is **255.255.255.255** , the IPv4 limited broadcast address.

- Configure IP helper addresses on the AP subnet router to forward broadcasts to one or more syslog servers.
- To reset the syslog host to **255.255.255.255** , use either the **default syslog host** or **no syslog host** command.
- To prevent the AP from sending syslog messages entirely, use **syslog host 0.0.0.0** .
- If a subnet contains more than 20 access points, do not log to the broadcast address. This prevents flooding the broadcast domain with log messages. Configure a specific syslog destination IP address.
- If you do not use the AP syslog feature, set the syslog host to **0.0.0.0** by entering the **syslog host 0.0.0.0** command.

Filter messages by severity

- Use the **syslog level <levelname>** command to filter messages based on severity.
- By default, the severity level is **informational** (**severity=6**), which means the system sends all messages except debugging logs to the server.

Filter messages by facility

- Use the **syslog facility <facilityname>** command to filter messages based on the facility code (facility name).
- By default, the facility is **kernel** (**kern** , **code=0**), so the system sends only kernel-related messages.
- To send messages from all facilities, configure the facility as **local7** .
- The configured facility name is included in the facility field of transmitted syslog messages.

Note Most AP log messages use the **kern** facility, while terminal access logs such as SSH and console access use the **auth** facility.

Secured syslog transmission

- Use the **syslog secured** command to enable Transport Layer Security (TLS), as defined in RFC 5425, to transmit syslog messages securely instead of using UDP.
- TLS-based syslog transmission is supported starting with software versions 17.9.6 and 17.12.1 (released in June 2023 and December 2023, respectively).

View syslog settings

To display the current syslog settings for an AP, use the **show capwap client configuration** command.

Configure a syslog server for an AP profile (CLI)

Configure a syslog server for an AP profile.

This ensures access points send system log messages to a specified remote server for central monitoring and troubleshooting.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile. Then, enter the AP profile configuration mode.

Example:

```
Device(config)# ap profile ap-profile
```

Step 3 Configure the facility parameter used by syslog messages.

Example:

```
Device(config-ap-profile)# syslog facility
```

Step 4 Enter the Syslog server IP address and parameters to define where the AP profile sends log messages.

Example:

```
Device(config-ap-profile)# syslog host ip-address
```

Step 5 Set the syslog server logging level to determine which messages the AP profile sends.

Example:

```
Device(config-ap-profile)# syslog level {alerts | critical | debugging | emergencies |
errors | informational | notifications | warnings}
```

Example:

```
Device(config-ap-profile)# syslog level
```

The syslog server supports these logging levels:

- **emergencies** —Signifies severity 0. Indicates that the system is not usable.
- **alerts** —Signifies severity 1. Indicates that an immediate action is required.
- **critical** —Signifies severity 2. Indicates a critical condition.
- **errors** —Signifies severity 3. Indicates an error condition.
- **warnings** —Signifies severity 4. Indicates a warning condition.
- **notifications** —Signifies severity 5. Indicates normal, but significant, conditions.
- **informational** —Signifies severity 6. Indicates informational messages.
- **debugging** —Signifies severity 7. Indicates debugging messages.

Note
Select a syslog level to enable it and all lower levels.

If you enable the *critical* syslog level, *alerts* and *emergencies* are also enabled.

Step 6 Return to privileged EXEC mode.

Example:

```
Device(config-ap-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The syslog server is configured for the specified AP profile. APs that use this profile will send log messages with the selected severity levels to the designated remote server.

Configure the AP syslog settings (GUI)

Set up syslog parameters for access points. This enables monitoring and centralized logging of events.

Procedure

Step 1 Choose **Configuration**, then **Tags and Profiles**, then **AP Join**.

Step 2 Select the APs from the AP list.

The **Edit AP Join Profile** window is displayed.

Step 3 Click the **Management** tab.

Step 4 Select the **Device** tab.

Step 5 In the **System Log** section:

a) From the **Facility Value** drop-down list, select a value.
b) Enter the IP address in the **Host IPv4 or IPv6 Address** field.
c) From the **Log Trap Value** drop-down list, select a value.
d) To enable **Secured**, check the box. To disable **Secured**, clear the box.

Step 6 Click **Update and Apply to Device**

The selected APs are updated with the new syslog configuration, enabling centralized logging according to your specified parameters.

Verify syslog server configurations

Verify global syslog server settings for all APs

To view the global syslog server settings for all access points that joins the controller, use this command:

```
Device# show ap config general
Cisco AP Name : APA0F8.4984.5E48
================================================

Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
```

```
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
```

```
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone FlexConnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```

Verify syslog server settings for a specific AP

To view the syslog server settings for a specific access point, use this command:

```
Device# show ap name <ap-name> config general
show ap name APA0F8.4984.5E48 config general
Cisco AP Name : APA0F8.4984.5E48
================================================

Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
```

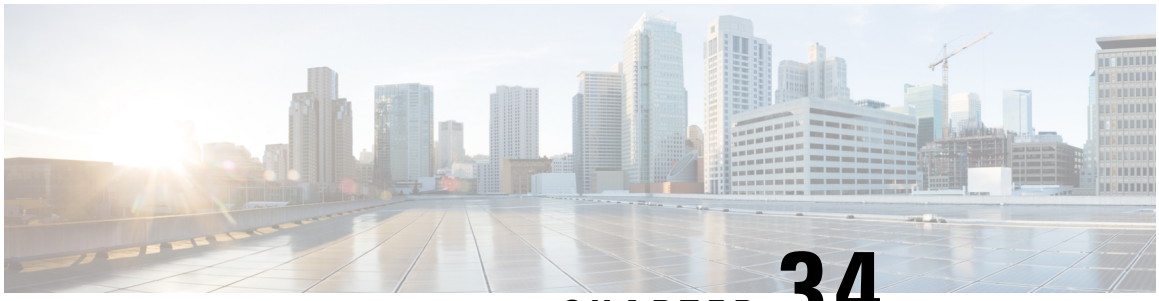
```
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone FlexConnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```



PART VI

Security

- IPv4 ACLs , on page 287
- DNS-Based Access Control Lists, on page 305
- Allowed List of Specific URLs, on page 319
- Web-Based Authentication , on page 323
- Central Web Authentication, on page 351
- ISE Simplification and Enhancements, on page 367
- Authentication and Authorization Between Multiple RADIUS Servers, on page 381
- Secure LDAP, on page 393
- RADIUS DTLS, on page 401
- MAC Filtering, on page 415
- Dynamic Frequency Selection, on page 421
- Managing Rogue Devices, on page 425
- Classifying Rogue Access Points, on page 453
- Configure Secure Shell , on page 463
- Private Shared Key, on page 471
- Multi-Preshared Key, on page 481
- Multiple Authentications for a Client, on page 491
- Locally Significant Certificates, on page 503



CHAPTER 34

IPv4 ACLs

- Network security with ACLs, on page 287
- How to configure ACLs, on page 295

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

An access control list (ACL) is a network security mechanism that

- defines permit and deny conditions for network packets
- filters inbound and outbound traffic as it traverses network devices, and
- enforces access and segmentation policies by controlling which packets are forwarded or dropped.

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. The controller checks each packet against the conditions you set in the access list, one at a time. The controller accepts or rejects the packets based on the first matching condition. Arrange your access list conditions carefully because the controller stops checking once it finds a match. If you do not set any restrictions, the controller forwards the packet. If you set restrictions and none are matched, the controller drops the packet. The controller can use ACLs on all packets it forwards . There is an implicit "deny any host" rule.

Configure access list

Configure access lists on the controller to provide basic security for your network. If you do not configure ACLs, the switch might allow all packets onto every part of your network.

Control access and traffic types

You can use ACLs to control which hosts access various parts of your network and to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow email traffic but block Telnet traffic.

Access control entries

An access control entry (ACE) is a rule component that

- specifies a *permit* or *deny* action for network packets
- defines conditions such as source or destination IP, protocol, or port that packets must meet to match the rule, and
- determines packet handling within an ordered ACL.

The meaning of *permit* or *deny* depends on the context in which the ACL is used.

Note The maximum number of ACEs that can be applied under an access policy (ACL) for central switching is 256 ACEs. For Flex Mode Local Switching, the maximum is 64 ACEs.

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.

ACEs and fragmented and unfragmented traffic

A fragmented or unfragmented packet is a type of network traffic that

- may be divided into multiple fragments as it traverses a network
- only includes Layer 4 information in the first fragment (such as TCP/UDP port, ICMP type/code), and
- requires specific ACE matching behavior depending on the presence or absence of Layer 4 information in its fragments.

IP packets can be fragmented as they cross the network. The first fragment, which contains the beginning of the packet, includes Layer 4 information such as TCP or UDP port numbers, or ICMP type and code. All other fragments lack Layer 4 information.

Some access control entries (ACEs) do not check Layer 4 information, so they can be applied to all packet fragments. ACEs that test Layer 4 information cannot be applied to most fragments in a fragmented IP packet. If a fragment contains no Layer 4 information and the ACE tests Layer 4 information, matching rules are modified.

- Permit ACEs that check Layer 3 information in the fragment, including protocol type such as TCP or UDP, match the fragment regardless of missing Layer 4 details.

Note For TCP ACEs with Layer 4 options, fragmented packets are dropped in accordance with RFC 1858.

- Deny ACEs that check Layer 4 information match a fragment only if 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, the *eq* keyword after the destination address checks if the TCP destination port matches the well-known port numbers for Simple Mail Transfer Protocol (SMTP) or Telnet.

- Packet A is a TCP packet from host 10.2.2.2, port 65000, sent to host 10.1.1.1 on the SMTP port. If the packet is fragmented, the first fragment matches the first ACE (a permit) because all Layer 4 information is included. The remaining fragments also match the first ACE, even if they do not include the SMTP port information, because the first ACE checks only Layer 3 information when applied to fragments. In this example, the fragment is a TCP packet sent to 10.1.1.1.
- Packet B is sent from host 10.2.2.2, port 65001, to host 10.1.1.2 on the Telnet port. If the packet is fragmented, the first fragment matches the second ACE (a deny) because it includes all Layer 3 and Layer 4 information. 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. As a result, packet B cannot be reassembled on host 10.1.1.2. Permitted fragments still use network bandwidth and resources on host 10.1.1.2 during the attempted reassembly.
- Packet C is sent from host 10.2.2.2, port 65001, to host 10.1.1.3 on the FTP port. This packet is fragmented. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because it does not check Layer 4 information. Layer 3 information in each fragment shows they are sent to host 10.1.1.3, while previous permit ACEs check different hosts.

Standard and extended IPv4 ACLs

An IPv4 access control list (ACL) is a packet-filtering feature that

- sequentially compares packets against a set of permit or deny conditions
- determines packet acceptance or rejection based on the first match, and
- enforces traffic control policies on network interfaces.

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 two types of ACLs 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 can include protocol-type information for more granular control.

Note Only extended ACLs are supported; standard ACLs are not supported.

IPv4 ACL switch unsupported features

Configuring IPv4 ACLs on this switch is similar to configuring IPv4 ACLs on other Cisco switches and routers.

The switch does not support the following ACL-related features:

- Non-IP protocol ACLs or bridge-group ACLs.
- IP accounting.
- Inbound and outbound rate limiting (except with QoS ACLs).
- Reflexive ACLs, URL redirect ACLs and dynamic ACLs are not supported. (except for some specialized dynamic ACLs used by the switch clustering feature).
- ACL logging for port ACLs and VLAN maps.

Access list numbers

The ACL number determines the type of access list you create.

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 14: 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

Access List Number	Type	Supported
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 create standard and extended named IP ACLs with the supported numbers. For example, you can name standard IP ACLs with numbers from 1 to 99, and extended IP ACLs with numbers from 100 to 199. With named ACLs, you can delete individual entries, but with numbered lists, you cannot.

Numbered standard IPv4 ACLs

When you create an ACL, an implicit deny statement is automatically present at the end. Any packet that does not match an earlier entry in the ACL is denied by this statement. For standard access lists, if you omit the mask in an IP host address ACL entry, the switch assumes a mask of 0.0.0.0.

- The switch rewrites standard access lists to move host match entries and entries with a don't care mask of 0.0.0.0 to the top of the list. Entries with non-zero don't care masks are placed below them. As a result, the ACEs may appear in a different order in show command output and in the configuration file than the order in which you entered them.
- After you create a numbered standard IPv4 ACL, you can apply it to VLANs, feature-id terminal lines (virtual teletype (VTY) lines), or interfaces.

Numbered extended IPv4 ACLs

A numbered extended IPv4 ACL is an access control mechanism that

- matches IPv4 traffic using both source and destination addresses
- optionally uses protocol type information for greater refinement, and
- requires that Access Control Entries (ACEs) be appended to the list since reordering or selective removal is unsupported.

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.

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:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**) and Open Shortest Path First routing (**ospf**)
- Internet Group Management Protocol (**igmp**) and Payload Compression Protocol (**pcp**)
- any Interior Protocol (**ip**) and Protocol-Independent Multicast (**pim**)
- IP in IP tunneling (**ipinip**) and Transmission Control Protocol (**tcp**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**) and 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, at times, not all commands that use IP access lists accept a named access 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.
- VLAN ACLs or VLAN maps are not supported.
- You can use standard or extended ACLs whether named or numbered, in VLAN maps.
- With IPv4 QoS ACLs, if you enter the class-map { match-all | match-any} class-map-name global configuration command, you can enter these match commands:
  1. **match access-group** *acl-name*



Note The ACL must be an extended named ACL.

  2. **match input-interface** *interface-id-list*
  3. **match ip dscp** *dscp-list*
  4. **match ip precedence** *ip-precedence-list*



Note You cannot enter the **match access-group** *acl-index* command.

ACL logging

ACL logging is a router logging feature that

- generates informational messages about packets permitted or denied by a standard IP access control list (ACL)
- records each message with details such as the action taken, the source IP address, and the access list number, and
- supports logging only for router ACLs (RACLs), not for ACLs used with Unicast Reverse Path Forwarding (uRPF).

The controller software can provide logging messages about packets permitted or denied by a standard IP access list. The level of messages logged to the console is controlled by the **logging console** commands, which control the syslog messages.

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 generates a logging message immediately. Subsequent packets are collected over 5-minute intervals before they are logged. The logging message includes the access list number, an indication of whether the packet was permitted or denied, the source IP address, and the number of packets from that source permitted or denied during the prior 5-minute interval.

Note The logging facility might drop some message packets if there are too many to handle, or if more than one logging message must be processed in a second. This behavior prevents the router from crashing due to too many logging packets. Therefore, do not use the logging facility as a billing tool or as an accurate source for 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.

The ACL scale for controllers is as follows:

- Cisco Catalyst 9800-40 wireless controller, Cisco Catalyst 9800-L wireless controller, Cisco Catalyst 9800-CL wireless controller (small and medium) support 128 ACLs with 128 access list entries (ACEs).
- Cisco Catalyst 9800-80 wireless controller and Cisco Catalyst 9800-CL wireless controller (large) support 256 ACLs and 256 ACEs.
- FlexConnect and Fabric mode APs support 96 ACLs.

Note If an ACL configuration cannot be implemented in the hardware due to an out-of-resource condition on the controller, then only the traffic in that VLAN arriving on that controller is affected. Software forwarding of packets might adversely impact the performance of the switch or switch stack, depending on the number of CPU cycles that this consumes.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the log keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

If a router ACL configuration cannot be applied in hardware, packets that arrive in a VLAN and must be routed are processed in software. However, packets are bridged in hardware. If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

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 access-lists hardware counters and show platform acl counters hardware privileged EXEC commands to obtain basic hardware ACL statistics for switched and routed packets.

Router ACLs operate in this way:

- 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, and if ip unreachables is disabled, the hardware drops flows that match a deny statement in a security ACL. 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 switched and routed in hardware.

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 and packets that 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 controller checks the packet against the ACL. If the ACL permits the packet, the controller continues to process the packet. If the ACL rejects the packet, the controller discards the packet.
- For outbound ACLs, after receiving and routing a packet to a controlled interface, the controller checks the packet against the ACL. If the ACL permits the packet, the controller sends the packet. If the ACL rejects the packet, the controller discards the packet.

If an undefined ACL has nothing listed in it, it is an empty access list.

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 half second per input interface. You can change this by using the ip icmp rate-limit unreachable global configuration command.

Restrictions for configuring IPv4 access control lists

General Network Security

These are restrictions for configuring network security with ACLs:

- A standard ACL and an extended ACL cannot have the same name.
- **AppleTalk** appears in the command-line help strings but is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- DNS traffic is permitted by default for clients that are awaiting web authentication, regardless of ACL entries.

IPv4 ACL Network Interfaces

These restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

How to configure ACLs

Configure IPv4 ACLs (GUI)

Define and apply IPv4 ACL rules for traffic management and security.

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 Click **Add**.

Step 3 In the **Add ACL Setup** dialog box, enter the details for these parameters.

- ACL Name: Enter the name for the ACL.
- ACL Type: IPv4 Standard.
- Sequence: Enter the sequence number.
- Action: Select ***Permit*** or ***Deny*** the packet flow from the drop-down list.

- Source Type: Choose ***any***, ***Host*** or ***Network*** from which the packet is sent.
- Log: Enable or disable logging.

Step 4 Click **Add**.

Step 5 Add the rest of the rules and click **Apply to Device**.

Configure IPv4 ACLs

Follow the procedure given below 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. .

Create a numbered standard ACL (GUI)

Define and apply a numbered standard ACL using the device's web interface using GUI.

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 On the **ACL** page, click **Add**.

Step 3 In the **Add ACL Setup** window, enter the details for these parameters.

- ACL Name: Enter the name for the ACL.
- ACL Type: IPv4 Standard.
- Sequence: Enter the sequence number.
- Action: Select **Permit** or **Deny** access from the drop-down list.
- Source Type: Choose **any** , **Host** or **Network**.
- Log: Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.

Step 4 Click **Add**.

Step 5 Click **Save & Apply to Device**.

Create a numbered standard ACL (CLI)

Follow the procedure given below to create a numbered standard ACL using commands:

Procedure

Step 1 Enable the privileged EXEC mode and enter your password if prompted.

Example:

```
Device# enable
```

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Define a standard IPv4 access list by using a source address and wildcard.

Example:

```
Device(config)# access-list 2 {deny | permit} source source-wildcard your_host
```

The *access-list-number* 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.

The *source* is the source address of the network or host from which the packet is being sent. Specify the source as:

- The 32-bit quantity in dotted-decimal format.
- The keyword **any** abbreviates for *source* and *source-wildcard* as 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard.
- The keyword **host** abbreviates source and *source-wildcard* as *source* 0.0.0.0.

The *source-wildcard* applies wildcard bits to the source address.

Note
Logging is supported only on ACLs attached to Layer 3 interfaces.

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 5 Verify your entries.

Example:

```
Device# show running-config
```

Step 6 (Optional) Save your entries in the configuration file.

Example:

```
Device# copy running-config startup-config
```

Create a numbered extended ACL (GUI)

Set up a numbered extended ACL to control network traffic based on source, destination, and protocol using the GUI.

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 On the **ACL** page, click **Add**.

Step 3 In the **Add ACL Setup** window, enter the details for these parameters.

- ACL Name: Enter the name for the ACL.
- ACL Type: IPv4 Extended.
- Sequence: Enter the sequence number.
- Action: Select **Permit** or **Deny** the packet flow from the drop-down list.
- Source Type: Choose **any** , **Host** or **Network** from which the packet is sent.
- Destination Type: Choose **any** , **Host** or **Network** to which the packet is sent.
- Protocol: Select a protocol from the drop-down list.
- Log: Enable or disable logging.
- DSCP: Enter to match packets with the DSCP value.

Step 4 Click **Add**.

Step 5 Click **Save & Apply to Device**.

Create a numbered extended ACL (CLI)

Follow the procedure given below to create a numbered extended ACL using commands:

Procedure

Step 1 Enters global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Defines an extended IPv4 access list and the access conditions.

Example:

```
Device# access-list 101 {deny | permit} ip host 10.1.1.2 any [precedence 0] [tos 0]
[fragments] log [log-input | smartlog] [time-range time-range-name] [dscp dscp]
```

The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699.

Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched.

For *protocol* , 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**.

Note
This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.

The *source* is the number of the network or host from which the packet is sent.

The *source-wildcard* applies wildcard bits to the source.

The *destination* is the network or host number to which the packet is sent.

The *destination-wildcard* applies wildcard bits to the destination.

Source, source-wildcard, destination, and destination-wildcard can be specified as:

- The 32-bit quantity in dotted-decimal format.
- The keyword **any** for 0.0.0.0 255.255.255.255 (any host).
- The keyword **host** for a single host 0.0.0.0.

The other keywords are optional and have these meanings:

- **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).
- **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.

  Note
  Your embedded controller must support the ability to:

  - Mark DCSP
  - Mark UP
  - Map DSCP and UP

    For more information on DSCP-to-UP Mapping , see:

    https://tools.ietf.org/html/draft-ietf-tsvwg-ieee-802-11-01

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**.

Step 3 Define an extended TCP access list and the access conditions.

Example:

```
Device(config)# access-list 101 {deny | permit} tcp source source-wildcard [operator-port]
 destination destination-wildcard 500 [established] [precedence 0] [tos 0] [fragments] log
 [log-input | smartlog] [time-range time-range-name] [dscp dscp] [flag]
```

The parameters are the same as those described for an extended IPv4 ACL, with these exceptions:

Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (range requires two port numbers separated by a space).

Enter the *port* 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.

The other optional keywords have these meanings:

- *flag*: 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 Define an extended UDP access list and the access conditions.

Example:

```
Device(config)# access-list 101 {deny | permit} udp source source-wildcard [operator port]
 destination destination-wildcard 100 precedence 0] [tos 0] [fragments] log [log-input |
smartlog] [time-range time-range-name] [dscp dscp
```

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 **flag** not valid for UDP.

Step 5 Define an extended ICMP access list and the access conditions.

Example:

```
Device(config)# access-list 101 {deny | permit} icmp source source-wildcard destination
destination-wildcard [ icmp-type | [[ icmp-type icmp-code ] | [ icmp-message ]] precedence
 0] [tos 0] [fragments] log [log-input | smartlog] [time-range time-range-name] [dscp dscp]
```

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:

- *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.
- *icmp-message*: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.

Step 6 (Optional) Define an extended IGMP access list and the access conditions.

Example:

```
Device(config)# access-list 101 {deny | permit} igmp source source-wildcard destination
destination-wildcard [ igmp-type ] precedence 0] [tos 0] [fragments] log [log-input |
smartlog] [time-range time-range-name] [dscp dscp]
```

The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.

igmp-type: 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**.

Step 7 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 8 (Optional) Save your entries in the configuration file.

Example:

```
Device# copy running-config startup-config
```

Create named standard ACLs (GUI)

Use this procedure to set up and apply ACLs for network security management within the device's GUI.

Procedure

Step 1 Click **Configuration** > **Security** > **ACL**.

Step 2 Click **Add** to create a new ACL setup.

Step 3 In the **Add ACL Setup** window, enter these parameters:

- ACL Name: Enter the name for the ACL.
- ACL Type: IPv4 Standard.
- Sequence: The valid range is between 1 and 99 or 1300 and 1999.
- Action: Select **Permit** or **Deny** access from the drop-down list.
- Source Type: Choose **any**, **Host** or **Network**.
- Log: Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.

Step 4 Click **Add** to add the rule.

Step 5 Click **Save & Apply to Device**.

Create named standard ACLs (CLI)

Define and apply named standard ACLs to permit or deny traffic from specified IPv4 source addresses using commands.

Procedure

Step 1 Enter the privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Define a standard IPv4 access list using a name, and enter the access-list configuration mode.

Example:

```
Device(config)# ip access-list standard 20
```

The name can be a number from 1 to 99.

Step 4 Specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped in access-list configuration mode. Use one of the these:

- **deny** { *source* [*source-wildcard*] | **host** *source* | **any** } [**log**]
- **permit** { *source* [*source-wildcard*] | **host** *source* | **any** } [**log**]

Example:

```
Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255
```

or

```
Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0
```

- **host** *source*: A source and source wildcard of *source* 0.0.0.0.
- **any**: A source and source wildcard of 0.0.0.0 255.255.255.255.

Step 5 Returns to privileged EXEC mode.

Example:

```
Device(config-std-nacl)# end
```

Step 6 Verify your entries.

Example:

```
Device# show running-config
```

Step 7 (Optional) Save your entries in the configuration file.

Example:

```
Device# copy running-config startup-config
```

Create extended named ACLs (GUI)

Follow these steps to filter traffic based on granular criteria, such as source/destination, protocol, or DSCP using the GUI.

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 Click **Add**.

Step 3 In the **Add ACL Setup** window, enter these parameters.

- ACL Name: Enter the name for the ACL.
- ACL Type: IPv4 Extended.
- Sequence: Enter the sequence number.
- Action: Select *Permit* or *Deny* the packet flow from the drop-down list.
- Source Type: Choose *any*, *Host* or *Network* from which the packet is sent.
- Destination Type: Choose *any*, *Host* or *Network* to which the packet is sent.
- Protocol: Select a protocol from the drop-down list.
- Log: Enable or disable logging.
- DSCP: Enter to match packets with the DSCP value.

Step 4 Click **Add**.

Step 5 Add the rest of the rules and click **Apply to Device**.

Create extended named ACLs (CLI)

Use this procedure to create extended named access control lists (ACLs) using CLI commands.

Procedure

Step 1 Enable the privileged EXEC mode and enter your password if prompted.

Example:

```
Device# enable
```

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Define an extended IPv4 access list using a name, and enter the access-list configuration mode.

Example:

```
Device(config)# ip access-list extended 150
```

The name can be a number from 100 to 199.

Step 4 Specify the conditions allowed or denied in access-list configuration mode { **deny** | **permit** } *protocol* { *source* [*source-wildcard*] | **host** *source* | **any** } { *destination* [*destination-wildcard*] | host *destination* | **any** } [**precedence** *precedence*] [**tos***tos*] [**log**] [**time-range** *time-range-name*]

Example:

```
Device(config-ext-nacl)# permit 0 any any
```

Use the **log** keyword to get access list logging messages, including violations.

- **host** *source*: A source and source wildcard of *source* 0.0.0.0.
- **host** *destintation*: A destination and destination wildcard of *destination* 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 Return to the privileged EXEC mode.

Example:

```
Device(config-ext-nacl)# end
```

Step 6 Verifiy your entries.

Example:

```
Device# show running-config
```

Step 7 (Optional) Save your entries in the configuration file.

Example:

```
Device# copy running-config startup-config
```

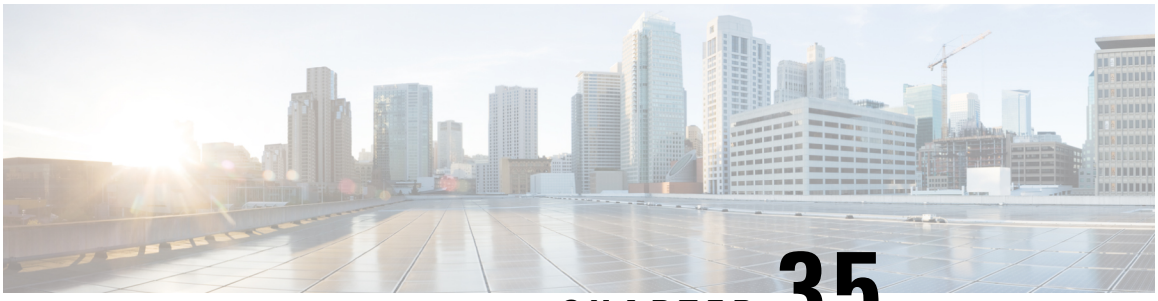
When you create extended ACLs, an implicit deny statement is added by default at the end of the ACL. If no match occurs before reaching the end, all traffic is denied. For standard ACLs, omitting the mask from an IP host address access list specification causes the system to use 0.0.0.0 as the mask.

New entries are always placed at the end of the ACL after you create it. You cannot add ACL entries at specific positions within an ACL. However, you can remove entries from a named ACL using the **no permit** and **no deny** access-list configuration mode commands.

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 you create a named ACL, apply it to interfaces or to VLANs .



CHAPTER 35

DNS-Based Access Control Lists

- DNS-based access control lists, on page 305
- URL filters, on page 309
- FlexConnect client IPv6 support with WebAuth pre and post ACL, on page 312

DNS-based access control lists

A DNS-based access control list is a wireless network security feature that

- allows administrators to permit or deny wireless client access to resources based on DNS domain names
- dynamically learns allowed or denied IP addresses through DNS snooping on access points, and
- enables pre-authentication and post-authentication URL filtering for enhanced client security.

DNS-based ACLs are used for wireless client devices. You can set pre-authentication ACLs on the embedded wireless controller to determine which data requests are allowed or blocked.

To enable DNS-based ACLs on the embedded wireless controller, configure the allowed or denied URLs for the ACLs. The URLs need to be pre-configured on the ACL.

During the registration phase, DNS-based ACLs allow the client to connect to the configured URLs. The embedded wireless controller is configured with the ACL name provided by the AAA server. If the AAA server returns the ACL name, the controller applies the ACL to the client for web redirection.

During client authentication, the AAA server returns the pre-authentication ACL (url-redirect-acl), which is the attribute name assigned to the ACL by the AAA server. The AP performs DNS snooping for each client until registration is complete and the client reaches the SUPPLICANT PROVISIONING state. When the embedded wireless controller receives the ACL configured with URLs, it sends a CAPWAP payload to the access point to enable DNS snooping for those URLs.

After URL snooping is enabled, the access point learns the IP address of the resolved domain name from the DNS response. If the domain name matches the configured URL, the access point parses the DNS response for the IP address and sends the IP address to the embedded wireless controller as a CAPWAP payload. The embedded wireless controller adds the IP address to the allowed list, which enables the client to access the configured URLs.

During pre-authentication or post-authentication, the DNS ACL is applied to the client in the AP. If the client roams from one AP to another AP, tthe DNS-learned IP addresses from the previous access point remain valid on the new AP.

Note Attach the URL filter to a policy profile for local mode. In FlexConnect mode, attach the URL filter to the flex profile; attaching it to a policy profile is not required.

Note DNS-based URLs require an active DNS query from the client. For URL filtering to work, configure DNS settings correctly.

Note The URL filter takes precedence over punt or redirect ACLs, as well as custom or static pre-authentication ACLs.

Restrictions on DNS-based access control lists

Restrictions for DNS-based ACLs

- In local modes, both Pre-authentication and Post-authentication filters are supported. In Flex (Fabric) mode, only the Pre-authentication filter is supported. ACL override pushed from ISE is not supported.
- FlexConnect Local Switching with External Web authentication, using URL filtering, is not supported until Cisco IOS XE Gibraltar 16.12.x.
- Fully qualified domain name (FQDN) ACLs and DNS-based ACLs are not supported on Cisco Wave 1 Access Points.
- The URL filter considers only the first 20 URLs, though you can add more. It uses regular regex patterns and permits wildcard characters only at the beginning or at the end of an URL. URL ACLs are defined and added to the FlexConnect policy profile, which is then associated with a WLAN. URL ACL creation follows a mechanism similar to local mode URL ACLs.
- In FlexConnect mode, the URL domain ACL works only when it is connected to a FlexConnect policy profile. The ACL can be attached to a WLAN by associating the policy profile with either the WLAN or local policies. However, you can override it using 'url-redirect-acl'. When the Cisco AV pair is received from ISE, the required policy for a client is pushed as part of the ADD MOBILE message.
- When an AP joins or when an existing URL ACL is modified and applied to a FlexConnect profile, the ACL definition, along with the mapped URL filter list, is pushed to the AP. The AP stores the URL ACL definition with the mapped ACL name and snoops DNS packets to learn the first IP address for each URL in the ACL. When the AP learns the IP addresses, it updates the controller of the URL and IP bindings. The controller records this information in the client database for future use.
- When a client roams to another AP during the pre-authentication state, the learned IP addresses are pushed to a new AP. Otherwise, these learned IP addresses are purged when a client enters the post-authentication state or when the TTL for the learned IP address expires.
-

Define ACLs

Extended ACLs are similar to standard ACLs, but they identify traffic more precisely.

- These commands allow you to define ACLs by name or by identification number.

```
Device(config)#ip access-list extended ?
<100-199> Extended IP access-list number
<2000-2699> Extended IP access-list number (expanded range)
WORD Access-list name
```

- This is the structure of a CLI ACL statement:

```
<sequence number> [permit/deny] <protocol> <address or any> eq <port number> <subnet>
<wildcard>
```

- For example:

```
1 permit tcp any eq www 192.168.1.0 0.0.0.255
```

The sequence number specifies where to insert the Access Control List Entry (ACE) in the order of ACEs within the ACL. You can define your statements using sequence numbers such as 10, 20, 30, 40, and so on.

The controller GUI allows you to write a complete ACL by going to the Configuration > Security > ACL page. You can view a list of protocols to choose from and make changes to an existing ACL.

Methods for applying ACLs

These are the ways to apply ACLs:

Security ACL: A security ACL defines the types of traffic to allow through the device and the types to block or drop. A security ACL can be applied in various scenarios:

- On SVI interfaces: The ACL is evaluated only against traffic routed through the interface.

```
Device(config)# interface Vlan<number>
```

```
Device(config-if)# ip access-group myACL in/out
```

- On a physical interface of the controller: The ACL is evaluated against all traffic passing through the interface. This method also restricts traffic on the controller management plane, along with SVI interface application.

```
Device(config)#interface GigabitEthernet1
```

```
Device(config-if)#ip access-group myACL in/out
```

In the wireless policy profile or WLAN. You can configure an ACL to apply to wireless client traffic in cases of central or local traffic switching. These ACLs are supported only in the inbound direction.

On the AP during FlexConnect local switching. The ACL is configured and applied from the policy profile on the controller. The ACL must be downloaded to the AP through the FlexConnect profile before it can be applied. Fabric mode APs in Software Defined Access also use Flex ACLs, even though these APs are not operating in FlexConnect mode.

A punt ACL or redirect ACL specifies which traffic is sent to the CPU for further processing instead of normal handling by the dataplane. For example, the Central Web Authentication (CWA) redirect ACL determines which traffic is intercepted and redirected to the web login portal. The ACL does not define traffic to drop or allow. It follows normal processing or forwarding rules and identifies what is sent to the CPU for interception.

A redirect ACL includes an implicit deny statement at the end. This security access list entry drops any traffic not explicitly allowed or sent to the CPU.

Configure ISE for central web authentication (GUI)

Perform these steps to configure ISE for Central Web Authentication using the GUI.

Procedure

Step 1 Login to the Cisco Identity Services Engine (ISE). Click **Policy** and then click **Policy Elements**. Click **Results**.

Step 2 Expand **Authorization** and click **Authorization Profiles**.

Step 3 Click **Add** to create a new authorization profile for URL filter.

Step 4 Enter a name for the profile in the **Name** field. For example, CentralWebauth.

Step 5 Select **ACCESS_ACCEPT** option from the **Access Type** drop-down list. Alternatively, in the **Common Tasks** section, check **Web Redirection**.

Step 6 Select the **Centralized Web Auth** option from the drop-down list. Specify the ACL and select the ACL value from the drop-down list.

Step 7 In the **Advanced Attributes Setting** section, select **Cisco:cisco-av-pair** from the drop-down list.

Note
Multiple ACL can be applied on the controller based on priority. In L2 Auth + webauth multi-auth scenario, if the ISE returns ACL during L2 Auth then ISE ACL takes precedence over the default webauth redirect ACL. This leads to traffic running in webauth pending state, if ISE ACL has permit rule. To avoid this scenario, you need to set the precedence for L2 Auth ISE returned ACL. The default webauth redirect ACL priority is 100. To avoid traffic issue, you need to configure the redirect ACL priority above 100 for ACL returned by ISE.

Step 8 Enter these one by one and click (+) icon after each of them:

- url-redirect-acl=<sample_name>
- url-redirect=<sample_redirect_URL>

  For example,

```
Cisco:cisco-av-pair = priv-lvl=15
Cisco:cisco-av-pair = url-redirect-acl=ACL-REDIRECT2
Cisco:cisco-av-pair = url-redirect=
https://9.10.8.247:port/portal/gateway?
sessionId=SessionIdValue&portal=0ce17ad0-6d90-11e5-978e-005056bf2f0a&daysToExpiry=value&action=cwa
```

Step 9 Verify contents in the **Attributes Details** section and click **Save**.

View DNS-based access control lists

To view details of a specified wireless URL filter, use this command:

```
Device# show wireless urlfilter details <urllist_flex_preauth>
```

To view the summary of all wireless URL filters, use this command:

```
Device# show wireless urlfilter summary
```

To view the URL filter applied to the client in the resultant policy section, use this command:

```
Device# show wireless client mac-address <MAC_addr> detail
```

Configuration examples for DNS-based access control lists

FlexConnect Mode

Example: Defining URL Filter List

This example shows how to define URL list in FlexConnect mode:

```
Device# configure terminal
Device(config)# urlfilter list urllist_flex_pre
Device(config-urlfilter-params)# action permit
Device(config-urlfilter-params)# redirect-server-ipv4 8.8.8.8
Device(config-urlfilter-params)# redirect-server-ipv6 2001:300:8::81
Device(config-urlfilter-params)# url url1.dns.com
Device(config-urlfilter-params)# end
```

Example: Applying URL Filter List to Flex Profile

This example shows how to apply an URL list to the FlexConnect profile in FlexConnect mode:

```
Device# configure terminal
Device(config)# wireless profile flex default-flex-profile
Device(config-wireless-flex-profile)# acl-policy acl_name
Device(config-wireless-flex-profile-acl)# urlfilter list urllist_flex_preauth
Device(config-wireless-flex-profile-acl)# end
```

URL filters

A URL filter is a wireless network access control feature that

- restricts access to websites based on specified URLs
- can operate before or after client authentication, and
- is applied together with an IP-based ACL for granular access control.

There are the two types of URL filters:

- Standard: Standard URL filters can be applied before client authentication (pre-auth) or after a successful client authentication (post-auth). Pre-auth filters are extremely useful for external web authentication because they allow access to the external login page and some internal websites before authentication takes place. After authentication, filters can block specific websites or allow only selected websites, while all other websites are blocked by default. After authentication, this type of URL filtering can be handled more flexibly by using Cisco DNS Layer Security (formerly called Umbrella). The standard URL filters apply the same action (permit or deny) for the whole list of URLs. It is either all permit or all deny.

- Enhanced: Enhanced URL filters allow you to specify a different action (deny or permit) for each URL in the list and include per-URL hit counters.

In both types of URL filters, you can use a wildcard sub-domain such as **`*.cisco.com`**. URL filters are standalone features but are always applied together with an IP-based ACL. A maximum of 20 URLs are supported in a given URL filter. Because one URL can resolve to multiple IP addresses, each client can track up to 40 resolved IP addresses. URL filters track only DNS records. The controller or APs do not track the resolved IP address if the DNS response uses a CNAME alias record.



Note If you have a POST-type URL filter and an ACL applied to a policy profile, the ACL blocks traffic to the URL if it does not include permit statements for the URLs. This issue can occur when the URL filter is POST with a permit statement but the ACL does not include a permit statement for the URLs. We recommend creating permit statements within the ACL for the IP addresses of the URLs rather than relying on the POST URL filter.

FlexConnect mode

Define URL filter list (CLI)

Define a URL filter list to permit or deny access to specific URLs and set up optional redirect servers for filtered traffic using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the URL filter list.

Example:

```
Device(config)# urlfilter list urllist_flex_preauth
```

Here, *list-name* refers to the URL filter list name. The list name must not exceed 32 alphanumeric characters.

Step 3 Configure the action: permit (allowed list) or deny (blocked list).

Example:

```
Device(config-urlfilter-params)# action permit
```

Step 4 Configure the IPv4 redirect server for the URL list.

Example:

```
Device(config-urlfilter-params)# redirect-server-ipv4 192.0.2.1
```

Here, *IPv4-address* refers to the IPv4 address.

Step 5 Configure the IPv6 redirect server for the URL list.

Example:

```
Device(config-urlfilter-params)# redirect-server-ipv6 2001:300:8::81
```

Here, *IPv6-address* refers to the IPv6 address.

Step 6 Configure an URL.

Example:

```
Device(config-urlfilter-params)# url url1.dns.com
```

Here, *url* refers to the name of the URL.

Step 7 Return to privileged EXEC mode.

Example:

```
Device(config-urlfilter-params)# end
```

Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Apply URL filter list to FlexConnect profile (CLI)

Use this procedure to enforce web access policies on FlexConnect APs through defined URL filter lists using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new flex policy.

Example:

```
Device(config)# wireless profile flex default-flex-profile
```

Step 3 Configure ACL policy.

Example:

```
Device(config-wireless-flex-profile)# acl-policy acl_name
```

Step 4 Apply the URL list to the Flex profile.

Example:

```
Device(config-wireless-flex-profile-acl)# urlfilter list urllist_flex_preauth
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-wireless-flex-profile-acl)# end
```

FlexConnect client IPv6 support with WebAuth pre and post ACL

FlexConnect client IPv6 support with WebAuth pre and post ACL

A WebAuth IPv6 ACL is an access control mechanism that

- enables an access point to enforce web authentication policies for IPv6 client traffic
- uses IOS IPv6 ACL definitions that are pushed to the AP in response to join events or policy changes, and
- differentiates pre-authentication and post-authentication access by assigning default or custom ACLs.

A change in the ACL policies of the FlexConnect profile can include adding a new ACL, deleting an ACL, or modifying an existing ACL.

ACL definitions are pushed to AP when any of these events occur:

- AP join.
- New ACL mapping in a new FlexConnect profile.
- Configuring IPv6 ACL definition in FlexConnect profile.

Default Local Web Authentication ACLs

The pre-defined default LWA IPv6 ACL is pushed to AP and assigned to the data plane.

Default External Web Authentication ACL

The default EWA ACLs are derived from the redirect portal address configured in the parameter map.

The types of default EWA ACLs include:

- Security ACL which is pushed and assigned to the AP.
- Intercept ACL which is pushed and assigned to the data plane.

FQDN ACL

- FQDN ACL is encoded along with IPv6 ACL and sent to AP.
- FQDN ACL is always a custom ACL.

These applies to FlexConnect and Local mode:

- If you are migrating from AireOS, you would explicitly need to execute these commands:

```
redirect append ap-mac tag ap_mac
redirect append wlan-ssid tag wlan
redirect append client-mac tag client_mac
```

- If the login page has any resource that needs to be fetched from the server, you will need to include those resource URLs in URL filtering.
- If you are trying to access an IPv6 URL and you have an IPv4 web server, the controller redirects the client to an internal page. Domain redirection is not supported in this context. It is recommended to have a dual-stack web server and configure virtual IPv6 address in the global parameter map.

Enable pre-authentication ACL for LWA and EWA (GUI)

Allow or restrict traffic before authentication on specified WLANs using the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.

Step 4 Choose **Security** > **Layer2** tab. Uncheck the **WPAPolicy**, **AES** and **802.1x** check boxes.

Step 5 Choose **Security** > **Layer3** tab. Select the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list and authentication list from the **Authentication List** drop-down list.

Step 6 Click **Show Advanced Settings** and under the **Preauthenticated ACL** settings, and select the IPv6 ACL from the **IPv6** drop-down list.

Step 7 Choose **Security** > **AAA** tab. Select the authentication list from the **Authentication List** drop-down list.

Step 8 Click **Apply to Device**.

Enable pre-authentication ACL for LWA and EWA (CLI)

Configure pre-authentication access control for WLAN client web authentication scenarios using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter the WLAN configuration sub-mode.

Example:

```
Device(config)# wlan wlan-demo 1 ssid-demo
```

- *wlan-name*: Enter the profile name. The range is from 1 to 32 alphanumeric characters.
- *wlan-id*: Enter the WLAN ID. The range is from 1 to 512.

- *SSID-name*: Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

  Note
  If you have already configured WLAN, enter **wlan wlan-name** command.

Step 3 Create a pre-authentication ACL for web authentication and disable the WPA security.

Example:

```
Device(config-wlan)# ipv6 traffic-filter web preauth_v6_acl
Device(config-wlan)# no security wpa
```

Step 4 Disable WPA2 ciphers for AES.

Example:

```
Device(config-wlan)#no security wpa wpa2 ciphers aes
```

Step 5 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)#no security wpa akm dot1x
```

Step 6 Configure web authentication.

Example:

```
Device(config-wlan)# security web-auth
```

Step 7 Enable authentication list for WLAN.

Example:

```
Device(config-wlan)# security web-auth authentication-list wcm_dot1x
```

Step 8 Map the parameter map.

Example:

```
Device(config-wlan)# security web-auth parameter-map param-custom-webconsent
```

Step 9 Shutdown the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Enable post-authentication ACL for LWA and EWA (GUI)

Enable post-authentication access control for LWA and EWA WLANs to enforce security policies after successful user authentication using the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**. The **Profile Name** is the profile name of the policy profile.

Step 4 Enter the **SSID** and the **WLAN ID**.

Step 5 Click **Apply to Device**.

Enable post-authentication ACL for LWA and EWA (CLI)

Configure security policies that restrict client access after authentication using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create policy profile for the WLAN.

Example:

```
Device(config)# wireless profile policy test1
```

The *profile-name* is the profile name of the policy profile.

Step 3 Create a named WLAN ACL.

Example:

```
Device(config-wireless-policy)# ipv6 acl testacl
```

Step 4 Return to the privileged EXEC mode.

Example:

```
Device(config-wireless-policy)# end
```

Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Enable DNS ACL for LWA and EWA (GUI)

Enable DNS-based access control for LWA and EWA WLANs by configuring the required settings using the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**. The **Profile Name** is the profile name of the policy profile.

Step 4 Enter the **SSID** and the **WLAN ID**.

Step 5 Click **Apply to Device**.

Enable DNS ACL for LWA and EWA (CLI)

Enable DNS ACLs to control DNS access for WLAN policy profiles that use LWA or EWA using commands.

Note Post-authentication DNS ACL is not supported.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create policy profile for the WLAN.

Example:

```
Device(config)# wireless profile policy test1
```

The *profile-name* is the profile name of the policy profile.

Step 3 Return to the privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Verify FlexConnect client IPv6 support with WebAuth pre and post ACL

To verify the client state after L2 authentication, use this command:

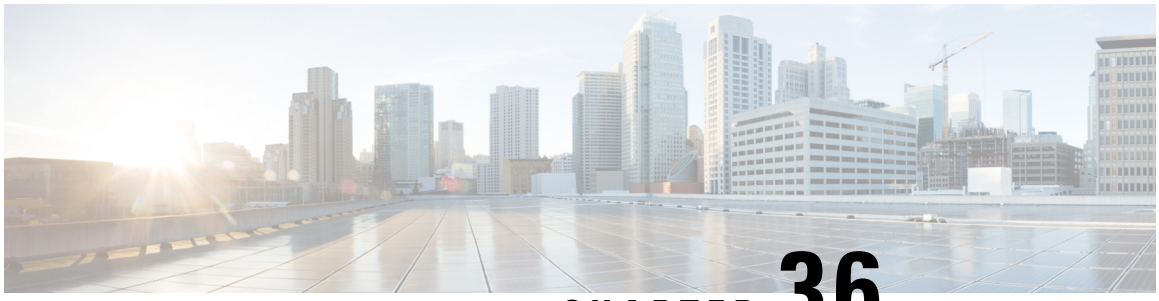
```
Device# show wireless client summary
Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State             Protocol Method
  Role
-------------------------------------------------------------------------------------------
1491.82b8.f8c1 AP4001.7A03.544C                 4      Webauth Pending   11n(5)   None
   Local
Number of Excluded Clients: 0
```

To verify the IP state, discovery, and MAC, use this command:

```
Device# show wireless dev da ip
IP                                      STATE       DISCOVERY   MAC
```

```
-------------------------------------------------------------------------------
15.30.0.4                                 Reachable   ARP         1491.82b8.f8c1
2001:15:30:0:d1d7:ecf3:7940:af60          Reachable   IPv6 Packet 1491.82b8.f8c1
fe80::595e:7c29:d7c:3c84                  Reachable   IPv6 Packet 1491.82b8.f8c1
```



CHAPTER 36

Allowed List of Specific URLs

- Allowed lists of specific URLs, on page 319

Allowed lists of specific URLs

An allowed list of specific URLs is a wireless controller feature that

- lets you add URLs to a list on the controller or access point so designated sites remain accessible
- enables web authentication use cases such as captive portals and walled gardens, and
- allows users to reach permitted sites before authenticating, even without full internet access.

Additional reference information

You do not need to authenticate to access URLs in the allowed list. If you try to reach a site not in the allowed list, you are redirected to the login page.

Accessing a support site

If "support.example.com" is added to the allowed list, users can access this site for help or information prior to logging into the network.

Add URL to allowed list

To configure a URL filter that allows specific URLs while blocking others.

This procedure is used in network environments where URL filtering is necessary to control access to web resources.

Before you begin

Ensure you have administrative access to the device and that URL filtering is supported.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the URL filter profile.

Example:

```
Device(config)# urlfilter list url-allowedlist-nbn
```

Step 3 Set the action for the URL filter list.

Example:

```
Device(config-urlfilter-params)# action permit
```

Configures the list as an allowed list, and the **deny** command configures the list as a blocked list.

Step 4 Configure the IP address of the redirect servers to which user requests will be redirected in case of denied requests.

Example:

```
Device(config-urlfilter-params)# redirect-server-ipv4 X.X.X.X
```

Step 5 Configure the URL to be allowed.

Example:

```
Device(config-urlfilter-params)# url www.cisco.com
```

Note **redirect-server-ipv4** and **redirect-server-ipv6** is applicable only in the local mode, specifically in post-authentication. For any further tracking or displaying any warning messages, the denied user request is redirected to the configured server.

But the **redirect-server-ipv4** and **redirect-server-ipv6** configurations do not apply to pre-authentication scenario as you will be redirected to the controller for the redirect login URL for any denied access.

You can associate the allowed URL with the ACL policy in flex profile.

Associating the allowed URL with the ACL policy in flex profile:

```
Device(config)# wireless profile flex default-flex-profile
            Device(config-wireless-flex-profile)# acl-policy user_v4_acl
            Device(config-wireless-flex-profile-acl)# urlfilter list url_allowedlist_nbn
            Device(config-wireless-flex-profile-acl)# exit
            Device(config-wireless-flex-profile)# description "default flex profile"

Device(config)# urlfilter enhanced-list urllist_pre_cwa
                Device(config-urlfilter-enhanced-params)# url url1.dns.com preference 1
action permit
                Device(config-urlfilter-enhanced-params)# url url2.dns.com preference 2
action deny
                Device(config-urlfilter-enhanced-params)# url url3.dns.com preference 3
action permit

Device(config)# wlan wlan5 5 wlan5
                    Device(config-wlan)#ip access-group web user_v4_acl
                    Device(config-wlan)#no security wpa
                    Device(config-wlan)#no security wpa
                    Device(config-wlan)#no security wpa wpa2 ciphers aes
```

```
Device(config-wlan)#no security wpa akm dot1x
Device(config-wlan)#security web-auth
Device(config-wlan)#security web-auth authentication-list default
Device(config-wlan)#security web-auth parameter-map global
Device(config-wlan)#no shutdown
```

Portal resolving to multiple IP addresses

You can use two IP addresses with the controller. However, the Web Auth Parameter Map allows pre-authentication to only one IP address. If an external portal, such as Cisco Spaces, resolves to multiple IP addresses or other HTTP resources require pre-authentication, you need a URL filter. The URL filter dynamically permits traffic to the configured URLs by snooping DNS requests and adding each resolved IP address to the intercept (redirect) and security (pre-auth) ACLs. This process enables you to access clients when necessary.

Additional information

- For FlexConnect deployments with local switching, apply the URL filter to the client at the AP for DNS snooping to work properly.
- Configuring the Web Auth Parameter Map automatically creates the following ACLs:
  - Redirect or intercept ACL (WA-v4-int): handles URL redirection.
  - Security ACL (WA-sec-): permits pre-auth access for HTTP or HTTPS, DNS, DHCP, and other required services.

Configuration steps for URL filter in FlexConnect profile

If you miss this step, the AP cannot dynamically add IP addresses. Clients may fail to redirect to the portal page when they request a secondary portal IP address.



Note Tip: Use visuals, diagrams, or tables as needed to clarify configuration mapping, ACL lists, or traffic flow.

Associating the allowed URL with the ACL policy in a Flex profile

```
  Device(config)# wireless profile flex default-flex-profile
  Device(config-wireless-flex-profile)# acl-policy WA-v4-<ip> (security ACL)
Device(config-wireless-flex-profile-acl)# urlfilter list url_allowedlist_nbn

  Device(config-wireless-flex-profile-acl)# exit
  Device(config-wireless-flex-profile)# description "default flex profile"
```

Verify URLs on the allowed list

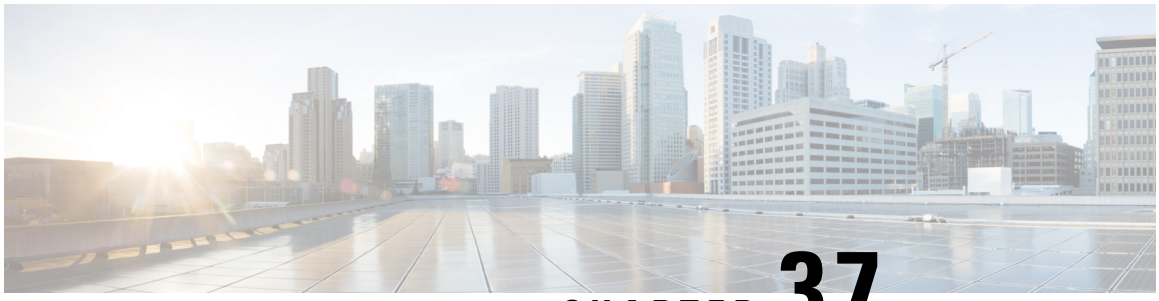
Verify URLs on the Allowed List.

```
Device# show wireless urlfilter summary
Black-list    - DENY
White-list    - PERMIT
Filter-Type   - Specific to Local Mode

URL-List                         ID  Filter-Type  Action   Redirect-ipv4  Redirect-ipv6
-------------------------------------------------------------------------------------------------
url-whitelist                    1    PRE-AUTH     PERMIT   192.0.2.1

Device#

Device# show wireless urlfilter details url-whitelist
List Name..................... : url-whitelist
Filter ID..................... : : 1
Filter Type................... : PRE-AUTH
Action........................ : PERMIT
Redirect server ipv4...... : 192.0.2.1
Redirect server ipv6...... :
Configured List of URLs
   URL....................... : www.example.com
```



CHAPTER 37

Web-Based Authentication

This chapter describes how to configure web-based authentication on the device. It contains these sections:

- Local web authentication , on page 323
- External Web Authentication, on page 341
- Authenticating sleeping clients, on page 346

Local web authentication

Local web authentication is a network security mechanism that

- authenticates users through a web browser login page
- enables access control on host systems that do not run IEEE 802.1X supplicants, and
- communicates with authentication, authorization, and accounting (AAA) servers to enforce security policies.

Feature history

Feature Name	Release	Description
Built-in Captive Portal Improvement	Cisco IOS XE 17.1.1	This release introduces support for special characters in the login portal banner title and banner text. The number of characters supported on the banner text has doubled to 400. The **exec-character-bits** command has been introduced.

Presentation options for local web authentication web pages

Local web authentication intercepts HTTP sessions on Layer 2 interfaces, and in some cases, Layer 3 interfaces (with restrictions for some switch models). When users try to access the network, local web authentication displays a login page and verifies user credentials with AAA servers, granting or denying access accordingly.

Local web authentication is categorized by the location where its web pages are hosted.

- **Internal**: Uses HTML pages (login, success, fail, and expire) stored on the embedded wireless controller .

- **Customized**: Uses customized HTML pages (login, success, fail, and expire) downloaded onto the embedded wireless controller for a customized user experience.
- **External**: Uses HTML pages hosted on an external web server.

We recommend that you follow the Cisco guidelines to create a customized web authentication login page. If you use the latest versions of Google Chrome or Mozilla Firefox browsers, ensure that your webauth bundle uses this line in the *login.html* file:

```
<body onload="loadAction();">
```

Web authentication modes

The types of web authentication differ according to the available web authentication pages.

- **Webauth**—The embedded wireless controller displays a page with the user name and password. Users enter valid credentials to gain network access.
- **Consent or web-passthrough**—The controller presents a policy page with the **Accept** and **Deny** buttons. Users simply click **Accept** to access the network – no credentials are required.
- **Webconsent**—This mode combines the features of Webauth and Consent. The embedded wireless controller displays a policy page with Accept or Deny buttons along with user name or password. Users must enter the correct credentials and click **Accept** to access the network.

Additional reference information

- You can view the webauth parameter-map information using the **show running-config** command.
- If you see tracebacks during client authentication, performance and behavior are not affected. Tracebacks may occur if Flexible Forwarding Mode (FFM) replies to Endpoint Profiler Module (EPM) for ACL application after the session is dequeued, usually because a timer expired or the session became **unauthorized**.
- You can configure web-based authentication on Layer 2 and layer 3 interfaces.
- When a client reaches the maximum limit of 200 HTTP connections, the system resets TCP connections and excludes your client from the network.

How local web authentication works

Summary

authentication enables secure network access for clients by prompting users to authenticate through a web login page. The process coordinates actions between the user, authentication server, and the network switch. The key components are:

- **User**: Initiates the HTTP session and enters authentication credentials.
- **Network switch**: Intercepts traffic, presents login pages, applies policies, and communicates with the authentication server.
- **Authentication server**: Verifies credentials and provides policy enforcement details.

Workflow

These are the stages of the process.

1. **Session initiation**: The user starts an HTTP session by attempting to access the network.
2. **Traffic interception and login page presentation**: The network switch intercepts the HTTP request and triggers the authorization process. It presents a login page for the user to enter their username and password.
3. **Credential submission and authentication**: After the user submits credentials, the switch forwards them to the authentication server
4. **Authentication outcome:**
   - If authentication succeeds, the switch downloads and activates the access policy assigned to the user from the server, then displays a login success page.
   - If authentication fails, the switch displays a login failure page. After a maximum number of failures, the login expired window appears, and the host is put on a watch list. Once the timeout expires, the user may try again
5. **Server non-response handling**: If the authentication server does not respond, and an AAA fail policy is in place, the switch applies the failure policy to the host and displays a login success page.
6. **Reauthentication triggers**: The switch reauthenticates a client if the host does not respond to an ARP probe (Layer 2) or does not send traffic within an idle timeout (Layer 3)

   suppress-feature-id="uabu_2960l_sw"

   .
7. **Session timeout enforcement**: The switch applies either the session timeout configured locally or provided by the server. The default local web authentication session timeout on controller is 1800 seconds from Cisco IOS XE 16.1.1 and later. The default session timeout value was infinite seconds, prior to Cisco IOS XE Denali 16.1.1.
8. **Session termination:**
   - If the terminate action is set to RADIUS, the switch sends a nonresponsive host request to the server; the server's response dictates the next action.
   - If the terminate action is set to default, the switch dismantles the session and removes the access policy.

Result

If authentication succeeds and the correct policies are applied, the user gains network access. If authentication fails, failure policies or timeout mechanisms may deny access.

Restrictions

- You cannot configure bypass authentication with the wireless web authentication feature.
- To update the redirect login URL, enable and then disable the WLAN. The redirect login URL specified in the web authentication parameter map changes when an AP rejoins.
- If authentication fails, users receive a failure page and can try to log in again. If the number of allowed attempts is exceeded, users may be excluded and receive a specific reason for the exclusion.

- Use the local web authentication feature to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

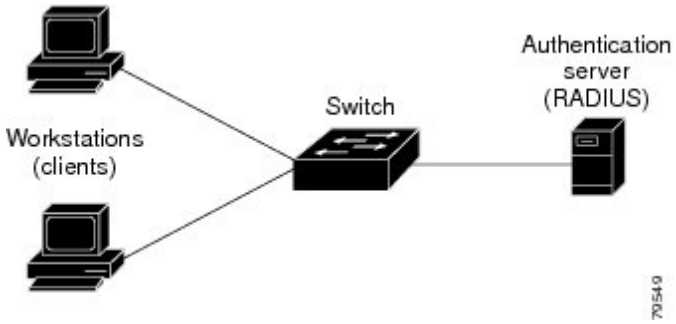
Roles of devices in local web authentication

In a local web authentication scenario, network devices assume specific roles to manage authentication and access to the LAN.

- **Client**: A device, such as a workstation, that requests network access and responds to authentication requests from the switch. The client must have an HTML browser with JavaScript enabled.
- **Authentication server**: A server that validates the identity of the client. The authentication server notifies the switch if the client is allowed or denied access to the LAN and related services.
- **Switch**: A network device that manages physical access to the network based on the authentication status of the client. The switch relays identity information and authorization responses between the client and the authentication server.

These device roles work together to ensure secure access control through local web authentication processes.

Figure 4: Local web authentication device roles



Banner messages for local web authentication

Local web authentication banners provide visual feedback to users during authentication on switches. The banners can display default or customized messages. They may also include branding or additional information on the login and result screens.

Default banner messages

When web authentication is enabled, one of these default messages appears on both the login screen and the authentication result pop-up page:

- Authentication Successful
- Authentication Failed
- Authentication Expired

Commands to configure local web authentication banner

You can configure the local web authentication banner using the .

Use the command in the global configuration mode:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner ?
file <file-name>
text <Banner text>
title <Banner title>
```

You can add a custom message, such as a switch, router, or company name, to the banner using the command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner text <text>
```

You can add a logo or text file to the banner using the command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner text <filepath>
```

Banner examples

Figure 5: Customized web banner: Authentication Successful

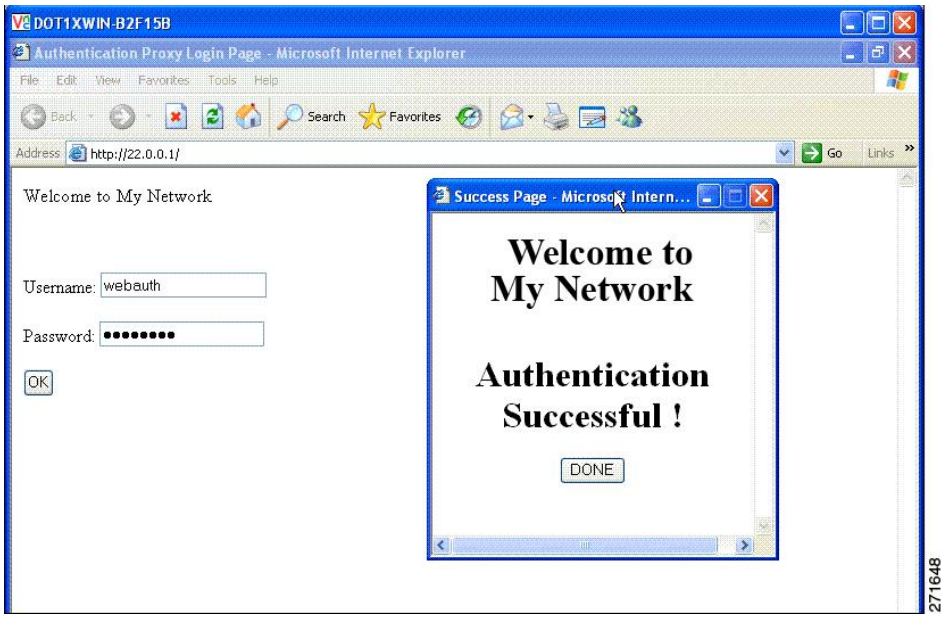
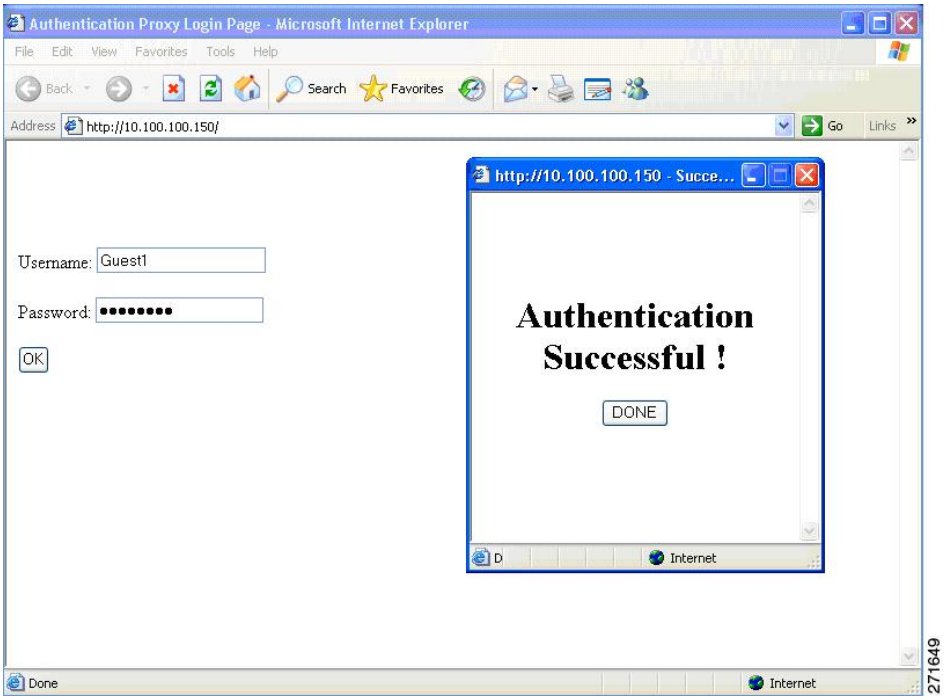


Figure 6: Login screen with no banner



Banner usage and behavior

- The default banners are *Cisco Systems* and *Switch host-name Authentication*. These banners appear on the login page. The *Cisco Systems* banner also appears on the authentication result pop-up page.
- The default banners appear unless custom banners are configured.
- If you do not enable a banner, only username and password dialog boxes appear on the web authentication login screen. You will not see a banner when you log into the switch.

Authentication states in customized local web authentication

During local web authentication, the switch's internal HTTP server hosts four HTML pages that communicate authentication states to the client.

You can replace the default internal HTML pages or specify a URL that redirects users after authentication. This URL replaces the internal Success page.

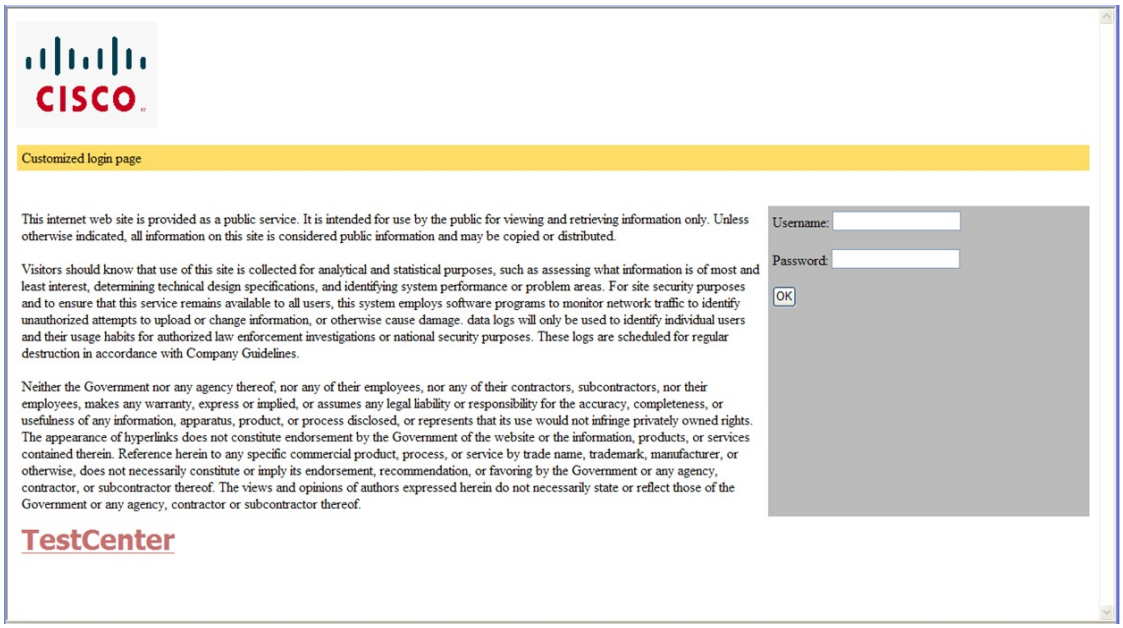
The four authentication pages and states are:

- Login: Credentials are requested from the client.
- Success: The client has authenticated successfully.
- Fail: Authentication attempt failed.
- Expire: The session expires after excessive failures.

You must configure all four pages. You can specify text or add a logo to each of the four pages.

Banner examples

Figure 7: Customizable Authentication Page



Best practices for customizing web authentication pages

- Add appropriate text to the banner and login pages as needed.
- Always include a valid HTML redirect command in the success page to redirect users to a specific URL after login
- Ensure the URL string is well-formed (for example, "http://www.cisco.com") to avoid browser errors.
- If you configure web pages for HTTP authentication, include the appropriate HTML commands. For example, HTML commands to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice.
- You can copy the configured web pages to the switch boot flash or the flash.
- The login page can reside on one flash device, while the success and failure pages can be stored on another flash device
- You must configure all four pages.
- All logo files—including image, flash, audio, video, and similar file types—stored in the system directory must use *web_auth_<filename>* as the file name. System directory examples are flash, disk0, or disk.
- You can copy the configured web pages to the switch boot flash or the flash.

Restrictions for customizing web authentication pages

- The banner page has no effect if it is configured together with a web authentication page.
- When the configured login form is enabled, the CLI command for redirecting users to a specific URL is unavailable. Configure redirection in the web page.

- If you enter the CLI command to redirect users to a specific URL after authentication and then configure web pages, the redirect command does not take effect.
- The configured authentication proxy feature supports both HTTP and SSL.

Guidelines for configuring a redirection URL for Successful Login page

- If you enable the custom authentication proxy web pages feature, you cannot use the redirection URL feature in the CLI. To redirect users after login, configure redirection in the custom login success page.
- If you enable the redirection URL feature, the configured authorization proxy banner is not be used.
- To remove the specification of a redirection URL, use the **no** form of the command.
- If a redirection URL is required after successful authentication, it must begin with a valid protocol prefix (such as http://) followed by the URL. If http:// is omitted, the browser might show a page not found error or similar issue

How to configure local web authentication

-

Configure default local web authentication

Table 15: Default local web authentication configuration

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	• None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Configure AAA authentication (GUI)

Set up AAA authentication to control network access using the GUI.

Use this task to define and assign authentication methods and server groups for device access.

Before you begin

Confirm that server groups are configured if you plan to use them.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**.

Step 2 In the **Authentication** section, click **Add**.

Step 3 In the **Quick Setup: AAA Authentication** window that is displayed, enter a name for your method list.

Step 4 Choose the type of authentication you want to perform before allowing access to the network, in the **Type** drop-down list.

Step 5 In the **Group** Type drop-down list, select either a group of servers as your access server or a local server to authenticate access.

Step 6 To configure a local server to act as a fallback method when servers in the group are unavailable, check the **Fallback** to local check box.

Step 7 Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click > icon to move them to the **Assigned Server Groups** list.

Step 8 Click **Save & Apply to Device**.

The controller is now configured with AAA authentication, and users are authenticated using the chosen methods.

Configure AAA authentication (CLI)

Enable authentication, authorization, and accounting (AAA) on the device to manage login methods and network access.

AAA centralizes user authentication for device access and network services. Select a named method list or the default list, depending on your VTY line configuration.

If no method list is configured under VTY lines, add the default method list 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+
```



Note Use the default list for AAA authorization if you plan to use features such as dACL.

Before you begin

Ensure that the TACACS+ server address and group name are available.

Procedure

Step 1 Enable AAA functionality.

Example:

```
Device
(config)# aaa new-model
```

Step 2 Define the list of authentication methods at login using the **aaa authentication login {default |** *named_authentication_list*} **group** *AAA_group_name* command.

Example:

```
Device(config)# aaa authentication login default group AAA_group_name
```

named_authentication_list refers to any name that is not greater than 31 characters.

AAA_group_name refers to the server group name. Define the **server_name** at the beginning of the configuration.

Step 3 Create an authorization method list for web-based authorization using the **aaa authorization network {default |** *named_authorization_list*} **group** *AAA_group_name* command.

Example:

```
Device(config)# aaa authorization network default group group1
```

Step 4 Specify a AAA server.

Example:

```
Device(config)# tacacs-server host 10.1.1.1
```

Configure the HTTP or HTTPS server (GUI)

Set up HTTP and HTTPS access to enable secure web-based management of the device.

HTTP provides basic web access; HTTPS secures connections with SSL encryption. You can configure access ports, authentication, trust points, and session policies to meet your security requirements.

Before you begin

Confirm necessary certificates are available if using HTTPS trust points.

Procedure

Step 1 Choose **Administration** > **Management** > **HTTP/HTTPS/Netconf**.

Step 2 In the **HTTP/HTTPS Access Configuration** section, enable HTTP Access and enter the port that will listen for HTTP requests. The default port is 80. Valid values are 80, and ports between 1025 and 65535.

Step 3 Enable **HTTPS Access** on the device and enter the designated port to listen for HTTPS requests. The default port is 1025. Valid values are 443 and ports between 1025 and 65535. Data exchanged over a secure HTTP connection is encrypted before transmission across the Internet. SSL encryption over HTTP enables secure configuration of the switch through a web browser.

Step 4 Choose the **Personal Identity Verification** as enabled or disabled.

Step 5 In the **HTTP Trust Point Configuration** section, enable **Enable Trust Point** to use Certificate Authority servers as trustpoints.

Step 6 From the **Trust Points** drop-down list, choose a trust point.

Step 7 In the **Timeout Policy Configuration** section, enter the HTTP timeout policy in seconds. Enter a value between one to 600 seconds.

Step 8 Enter the number of minutes of inactivity allowed before the session times out. Enter a value between 180 and 1200 seconds.

Step 9 Enter the server life time in seconds. Enter a value between one and 86400 seconds.

Step 10 Enter the maximum number of requests the device can accept. Valid values range from one and 86400 requests.

Step 11 Save the configuration.

The device is configured for HTTP or HTTPS access according to your specified settings.

Configure the HTTP server (CLI)

Enable HTTP or HTTPS server functionality on your device to support local web authentication.

Local web authentication requires the HTTP server to be enabled on the controller.

You can configure both HTTP and HTTPS servers to support device management and user authentication.

Both HTTP and HTTPS servers can be configured to support device management and user authentication.

Note that some browsers, such as the Apple psuedo-browser, do not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Procedure

Step 1 Enable the HTTP server. The local web authentication feature uses the HTTP server to communicate with the hosts for user authentication.

Example:

```
Device(config)# ip http server
```

Step 2 Enable HTTPS.

Example:

```
Device(config)# ip http secure-server
```

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 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The controller is now configured with the HTTP or HTTPS or both. This enables local web authentication and secure browser access for users.

Configure the maximum web authentication request retries

Set the maximum number of web authentication request retries to define how many times the system attempts authentication before it stops.

Configure this setting to adjust the tolerance for failed web authentication attempts, which can improve network security and user experience.

Before you begin

Ensure you are in global configuration mode on your device.

Procedure

Step 1 Configure the maximum web authentication request retries.

Example:

```
Device(config)# wireless security web-auth retries  number
```

number is the maximum number of web authentication request retries. The valid range is 0-20.

Step 2 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

You have configured the specified maximum number of web authentication request retries.

Configure a local banner in web authentication page (GUI)

Present a custom banner to users on the web authentication page to meet organizational messaging or compliance requirements.

Use this task to configure a banner that will display on the login page for web authentication sessions.

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 In the **General** tab and choose the required Banner Type:

- If you choose **Banner Text**, enter the required banner text to be displayed.
- If you choose **File Name**, specify the path of the file from which the banner text has to be picked up.

Step 4 Click **Update & Apply**.

The specified banner displays on the web authentication page as configured.

Configure a local banner in web authentication page (CLI)

Present a custom banner to users on the Web Authentication page to meet organizational messaging or compliance requirements.

Use this task to configure a banner that displays on the login page for web authentication sessions.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the web authentication parameters. Enters the parameter map configuration mode.

Example:

```
Device(config)# parameter-map type webauth param-map
```

Step 3 Enable the local banner using the **banner** [*file* | *banner-text* | *title*] command.

Example:

```
Device(config-params-parameter-map)# banner http C My Switch C
```

Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file* that indicates a file (for example, a logo or text file) that appears in the banner, or *title* that indicates the title of the banner.

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config-params-parameter-map)# end
```

The specified banner displays on the web authentication page as configured.

Configure trustpoint for local web authentication

Configure a trustpoint on your controller to support secure local web authentication.

Set up a trustpoint so the controller can present a domain-specific certificate. Client browsers recognize and accept this certificate during redirection to the portal (for example, example.com).

Before you begin

Ensure that a certificate is installed on your embedded wireless controller.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create the parameter map.

Example:

```
Device(config)# parameter-map type webauth global
```

Step 3 Configure trustpoint for local web authentication.

Example:

```
Device(config-params-parameter-map)# trustpoint trustpoint-name
```

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config-params-parameter-map)# end
```

During local web authentication, the controller presents the specified domain-specific certificate. Clients' browsers then recognize and accept the portal redirection.

Configuration examples for local web authentication

-

Example: obtain a web authentication certificate

This example shows how to obtain web authentication certificate.

```
DeviceDevice# configure terminal
Device(config)# crypto pki import cert pkcs12 tftp://10.1.0.100/ldapserver-cert.p12 cisco
Device(config)# end
Device# show crypto pki trustpoints cert
 Trustpoint cert:
    Subject Name:
```

```
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
          Serial Number (hex): 00
    Certificate configured.
Device# show  crypto pki certificates cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    Name: ldapserver
    e=rkannajr@cisco.com
    cn=ldapserver
    ou=WNBU
    o=Cisco
    st=California
    c=US
  Validity Date:
    start date: 07:35:23 UTC Jan 31 2012
    end   date: 07:35:23 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12
  Storage: nvram:rkannajrcisc#4.cer

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 00
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Validity Date:
    start date: 07:27:56 UTC Jan 31 2012
    end   date: 07:27:56 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12 ldap
  Storage: nvram:rkannajrcisc#0CA.cer
```

Example: display a web authentication certificate

This example shows how to display a web authentication certificate.

```
Device# show crypto ca certificate verb
     Certificate
     Status: Available
     Version: 3
     Certificate Serial Number (hex): 2A9636AC00000000858B
     Certificate Usage: General Purpose
     Issuer:
    cn=Cisco Manufacturing CA
    o=Cisco Systems
    Subject:
    Name: WS-C3780-6DS-S-2037064C0E80
    Serial Number: PID:WS-C3780-6DS-S SN:FOC1534X12Q
    cn=WS-C3780-6DS-S-2037064C0E80
    serialNumber=PID:WS-C3780-6DS-S SN:FOC1534X12Q
    CRL Distribution Points:
    http://www.cisco.com/security/pki/crl/cmca.crl
    Validity Date:
    start date: 15:43:22 UTC Aug 21 2011
    end   date: 15:53:22 UTC Aug 21 2021
    Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
    Signature Algorithm: SHA1 with RSA Encryption
    Fingerprint MD5: A310B856 A41565F1 1D9410B5 7284CB21
    Fingerprint SHA1: 04F180F6 CA1A67AF 9D7F561A 2BB397A1 0F5EB3C9
    X509v3 extensions:
    X509v3 Key Usage: F0000000
      Digital Signature
      Non Repudiation
      Key Encipherment
      Data Encipherment
    X509v3 Subject Key ID: B9EEB123 5A3764B4 5E9C54A7 46E6EECA 02D283F7
    X509v3 Authority Key ID: D0C52226 AB4F4660 ECAE0591 C7DC5AD1 B047F76C
    Authority Info Access:
    Associated Trustpoints: CISCO_IDEVID_SUDI
    Key Label: CISCO_IDEVID_SUDI
```

Example: choose the default web authentication login page

This example shows how to choose a default web authentication login page.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
This operation will permanently convert all relevant authentication commands to their CPL
control-policy equivalents. As this conversion is irreversible and will
disable the conversion CLI 'authentication display [legacy|new-style]', you are strongly
advised to back up your current configuration before proceeding.
Do you wish to continue? [yes]: yes
Device(config)# wlan wlan50
Device(config-wlan)# shutdown
Device(config-wlan)# security web-auth authentication-list test
Device(config-wlan)# security web-auth parameter-map test
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show running-config | section wlan50
wlan wlan50 50 wlan50
```

```
 security wpa akm wpa2
 security wpa wpa1
 security wpa wpa1 ciphers aes
 security wpa wpa1 ciphers tkip
 security web-auth authentication-list test
 security web-auth parameter-map test
 session-timeout 1800
 no shutdown

Device# show running-config | section parameter-map type webauth test
parameter-map type webauth test
 type webauth
```

Example: choosing a CWA login page from an IPv4 external web server

This example shows how to choose a customized web authentication login page from an IPv4 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv4 192.0.2.1.
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9.1.0.100/login.html
Device(config-params-parameter-map)# redirect portal ipv4 9.1.0.100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv4 192.0.2.1.
parameter-map type webauth test
type webauth
redirect for-login http://10.1.0.100/login.html
redirect portal ipv4 10.1.0.100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: choose a customized web authentication login page from an IPv6 external web server

This example shows how to choose a customized web authentication login page from an IPv6 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv6 2001:DB8::/48
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9:1:1::100/login.html
Device(config-params-parameter-map)# redirect portal ipv6 9:1:1::100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv6 2001:DB8::/48
parameter-map type webauth test
type webauth
redirect for-login http://10:1:1::100/login.html
redirect portal ipv6 10:1:1::100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: assigning login, login failure, and logout pages per WLAN

This example shows how to assign login, login failure and logout pages per WLAN.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
Device(config-params-parameter-map)# custom-page login device flash:loginsantosh.html
Device(config-params-parameter-map)# custom-page login expired device flash:loginexpire.html
Device(config-params-parameter-map)# custom-page failure device flash:loginfail.html
Device(config-params-parameter-map)# custom-page success device flash:loginsucess.html
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://10.1.0.100/login.html
 redirect portal ipv4 10.1.0.100
 custom-page login device flash:loginsantosh.html
 custom-page success device flash:loginsucess.html
 custom-page failure device flash:loginfail.html
 custom-page login expired device flash:loginexpire.html
```

Example: configure preauthentication ACL

This example shows how to configure preauthentication ACL.

```
Device# configure terminal
Device(config)# wlan fff
Device(config-wlan)# shutdown
Device(config-wlan)# ip access-group web preauthrule
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show wlan name fff
```

Example: configure webpassthrough

This example shows how to configure webpassthrough.

```
Device# configure terminal
Device(config)# parameter-map type webauth webparalocal
Device(config-params-parameter-map)# type consent
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://10.1.0.100/login.html
 redirect portal ipv4 10.1.0.100
```

Verify web authentication type

To verify the web authentication type, run the command:

```
Device# show parameter-map type webauth all
Type Name
---------------------------------
Global global
Named webauth
Named ext
Named redirect
Named abc
```

```
Named glbal
Named ewa-2
```

```
Device# show parameter-map type webauth global
Parameter Map Name : global
Banner:
Text : CisCo
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Enabled
Sleeping-Client timeout : 60 min
Virtual-ipv4 : 10.0.2.1.
Virtual-ipv4 hostname :
Webauth intercept https : Disabled
Webauth Captive Bypass : Disabled
Webauth bypass intercept ACL :
Trustpoint name :
HTTP Port : 80
Watch-list:
Enabled : no
Webauth login-auth-bypass:
```

```
Device# show parameter-map type webauth name global
Parameter Map Name : global
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Disabled
Webauth login-auth-bypass:
```

External Web Authentication

Configure EWA with single WebAuth server address and default ports (80/443) (CLI)

Configure External Web Authentication (EWA) on your device to use a single WebAuth server address with default ports (80 or 443).

Use this procedure to redirect guest WLAN clients to a specific WebAuth portal using default HTTP or HTTPS ports.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Define the authentication method at login.

Example:

```
Device(config)# aaa authentication login WEBAUTH local
```

Step 3 Create the parameter map using the **parameter-map type webauth** *parameter-map-name* command.

Example:

```
Device(config)# parameter-map type webauth ISE-Ext-Webauth_IP
```

The *parameter-map-name* must not exceed 99 characters.

Step 4 Configure the webauth type parameter.

Example:

```
Device(config-params-parameter-map)# type webauth
```

Step 5 Configure the URL string for redirect during login using the **redirect for-login** *URL-String*command.

Example:

```
Device(config-params-parameter-map)#  redirect for-login
https://192.168.0.98:443/portal/PortalSetup.action?portal=ad64b062-1098-11e7-8591-005056891b52
```

Step 6 Configure the external portal IPv4 address.

Example:

```
Device(config-params-parameter-map)# redirect portal ipv4 192.168.0.98
```

Step 7 Return to global configuration mode.

Example:

```
Device(config-params-parameter-map)# exit
```

Step 8 Configure a WLAN using the **wlan** *wlan-name wlan-id SSID-name* command.

Example:

```
Device(config)#  wlan EWLC3-GUEST 3 EWLC3-GUEST
```

Step 9 Disable adaptive 11r.

Example:

```
Device(config-wlan)# no security ft adaptive
```

Step 10 Disable WPA security.

Example:

```
Device(config-wlan)# no security wpa
```

Step 11 Disable WPA2 security.

Example:

```
Device(config-wlan)# no security wpa wpa2
```

Step 12 Disable WPA2 ciphers for AES.

Example:

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

Step 13 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Step 14 Enable web authentication for WLAN.

Example:

```
Device(config-wlan)# security web-auth
```

Step 15 Enable authentication list for dot1x security using the **security web-auth authentication-list** *authenticate-list-name* command.

Example:

```
Device(config-wlan)# security web-auth authentication-list WEBAUTH
```

Step 16 Configure the parameter map using the **security web-auth parameter-map** *parameter-map-name* command.

Example:

```
Device(config-wlan)# security web-auth parameter-map ISE-Ext-Webauth_IP
```

Note
If parameter map is not associated with a WLAN, the configuration uses the global parameter map.

Step 17 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Once enabled for the configured WLAN, web authentication redirects clients to the specified WebAuth portal using default ports.

Configure EWA with multiple web servers and/or ports different than default (80/443)

Configure an External Web Authentication (EWA) workflow to support multiple web servers and custom port numbers using CLI.

Enable guest access using EWA if there are multiple external web servers or if web authentication must use ports other than the default ports of 80 or 443.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Define an extended IPv4 access list using a name, and enters access-list configuration mode.

Example:

```
Device(config)# ip access-list extended preauth_ISE_Ext_WA
```

Step 3 Permit access from any host to the first external web server port number 8443.

Example:

```
Device(config)# access-list-number permit tcp any host external_web_server_ip_address1 eq
port-number
```

Example:

```
Device(config)# 10 permit tcp any host 192.168.0.98 eq 8443
```

Step 4 Permit access from any host to the second external web server port number 8443.

Example:

```
Device(config)# access-list-number permit tcp any host external_web_server_ip_address2 eq
port-number
```

Example:

```
Device(config)# 10 permit tcp any host 192.168.0.99 eq 8443
```

Step 5 Permit DNS UDP traffic.

Example:

```
Device(config)# access-list-number permit udp any eq domain
```

Example:

```
Device(config)# 20 permit udp any any eq domain
```

Step 6 Permit DHCP traffic using the **bootpc**.

Example:

```
Device(config)# 30 permit udp any any eq bootpc
```

Example:

```
Device(config)# access-list-number permit udp any any eq bootpc
```

Step 7 Permit DHCP traffic using **bootps**.

Example:

```
Device(config)# access-list-number permit udp any any eq bootps
```

Example:

```
Device(config)# 40 permit udp any any eq bootps
```

Step 8 Permit the access from the first external web server port 8443 to any host.

Example:

```
Device(config)# access-list-number permit tcp any host external_web_server_ip_address1 eq
port-number any
```

Example:

```
Device(config)# 50 permit tcp host 192.168.0.98 eq 8443 any
```

Step 9 Permit the access from the second external web server port 8443 to any host.

Example:

```
Device(config)# access-list-number permit tcp any host external_web_server_ip_address2 eq
port-number any
```

Example:

```
Device(config)# 50 permit tcp host 192.168.0.99 eq 8443 any
```

Step 10 Permit DNS TCP traffic.

Example:

```
Device(config)# access-list-number permit tcp any eq domain
```

Example:

```
Device(config)# 60 permit tcp any any eq domain
```

Step 11 Deny all the other traffic.

Example:

```
Device(config)# access-list-number deny ip any any
```

Example:

```
Device(config)# 70 deny ip any any
```

Step 12 Create the WLAN using the **wlan** *wlan-name wlan-id ssid* command.

Example:

```
Device(config)# wlan wlan-name wlan-id ssid
```

Example:

```
Device(config)# wlan EWLC3-GUEST 3 EWLC3-GUEST
```

Step 13 Configure the IPv4 WLAN web ACL

Example:

```
Device(config-wlan)# ip access-group web name
```

The variable *name* specifies the user-defined IPv4 ACL name

Step 14 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

This configuration allows EWA with multiple external web servers and ports. It supports DNS or DHCP traffic and blocks unauthorized traffic.

Configure wired guest EWA with multiple web servers, ports different than default (80/443)

Configure Wired Guest External Web Authentication (EWA) when using multiple web servers or ports other than the default ports of 80 or 443 using CLI.

Wired Guest LAN profiles do not allow manual ACL assignment directly. To support multiple web servers or custom ports, use the bypass ACL in the global parameter map.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Define an extended IPv4 access list using a name, and enter access-list configuration mode.

Example:

```
Device(config)# ip access-list extended BYPASS_ACL
```

Step 3 Allow the traffic to switch centrally.

Example:

```
Device(config)# 10 deny ip any host 192.168.0.45
```

Step 4 Allow the traffic to switch centrally

Example:

```
Device(config)# access-list-number  deny ip any host hostname
```

Example:

```
Device(config)# 20 deny ip any host 4.0.0.1
```

Step 5 Creates a parameter map and enter parameter-map webauth configuration mode.

Example:

```
Device(config)# parameter-map type webauth global
```

Step 6 Create a WebAuth bypass intercept using the ACL name.

Example:

```
Device(config-params-parameter-map)# webauth-bypass-intercept BYPASS_ACL
```

Note
You cannot manually apply an ACL to the wired guest profile to configure external web authentication with multiple IP addresses or different ports. To work around this, use the bypass ACL in the wired guest profile.

Step 7 Return to privileged EXEC mode.

Example:

```
Device(config-params-parameter-map)# end
```

The wired guest profile uses the bypass ACL, enabling external web authentication with multiple web servers or custom ports.

Authenticating sleeping clients

A sleeping client is a wireless device that

- has completed web authentication and been granted guest access
- is allowed to enter sleep mode and wake up without reauthenticating through the login page, and
- the controller stores sleeping client information for a configurable duration before requiring reauthentication.

The valid range is 10 minutes to 43200 minutes, with the default being 720 minutes. You can also configure this duration on WebAuth parameter map that is mapped to a WLAN. The sleeping client timer is activated when conditions such as idle timeout, session timeout, WLAN disabling, or AP nonoperational status occur.



Caution If the MAC address of a client in sleep mode is spoofed, a fake device, such as a laptop, can be mistakenly authenticated.

Feature History

Feature Name	Release	Description
Webauth Sleeping Client Support	Cisco IOS XE 17.1.1s	The web authentication sleeping clients feature supports multiple combinations of authentications for a given client, which are configured on the WLAN profile.

Scenarios where sleeping clients do not need reauthentication

- Suppose there are two embedded wireless controller s in a mobility group. A client that is associated with one embedded wireless controller goes to sleep and then wakes up and gets associated with the other embedded wireless controller.
- Suppose there are three embedded wireless controller s in a mobility group. A client that is associated with the second embedded wireless controller that is anchored to the first controller goes to sleep, wakes up, and gets associated with the third embedded wireless controller.
- A client sleeps, wakes up and gets associated with the same or different export foreign embedded wireless controller that is anchored to the export anchor.

Guidelines for mobility scenarios

- This feature is supported on the FlexConnect scenario with local switching and central authentication.
- L2 roaming in the same subnet is supported.
- Anchor sleeping timer is applicable.
- The sleeping client information is shared between multiple auto anchors when a sleeping client moves from one anchor to another.

Restrictions on authenticating sleeping clients

- If the MAC address of a client in sleep mode is spoofed, a fake device, such as a laptop, can be mistakenly authenticated.
- The sleep client feature works only for WLAN configured with WebAuth security.
- You can configure the sleeping clients only on a per WebAuth parameter-map basis.

- The authentication of sleeping clients feature is supported only on WLANs that have Layer 3 security enabled.
- When Layer 3 security is enabled, the Authentication, Passthrough, and On MAC Filter failure web policies are supported. The Splash Page Web Redirect web policy is not supported.
- The central web authentication of sleeping clients is not supported.
- The authentication of sleeping clients feature is not supported on guest LANs and remote LANs.
- A guest access sleeping client that has a local user policy is not supported. In this case, the WLAN-specific timer is applied.

Configure authentication for sleeping clients (GUI)

Enable authentication for sleeping clients to ensure devices can maintain secure network access even after entering sleep mode.

Sleeping clients, such as laptops or mobile devices, periodically enter low-power states. You configure authentication to ensure they reconnect securely and seamlessly when they wake.

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 Check **Sleeping Client Status** check box.

Step 4 Click **Update & Apply to Device**.

Authentication for sleeping clients is now enabled. Devices entering sleep mode will reconnect securely without manual intervention.

Configure authentication for sleeping clients (CLI)

Configure authentication parameters to manage sleeping wireless clients and control their session persistence on the device.

Sleeping clients are wireless devices that temporarily disconnect due to power-saving or roaming behaviors. Configuring authentication for sleeping clients ensures they are recognized and handled appropriately during these periods.

Procedure

Step 1 Create a parameter map and enters parameter-map webauth configuration mode.

Example:

```
Device(config)# parameter-map type webauth global
```

Step 2 Configure the sleeping client timeout to 100 minutes.

Example:

```
Device(config-params-parameter-map)# sleeping-client timeout 100
```

Valid range is between 10 minutes and 43200 minutes.

Note
If you do not use the timeout keyword, the sleeping client is configured with the default timeout value of 720 minutes.

Step 3 Exit parameter-map webauth configuration mode and returns to privileged EXEC mode.

Example:

```
Device# end
```

Step 4 (Optional) Show the MAC address of the clients and the time remaining in their respective sessions.

Example:

```
Device# show wireless client sleeping-client
```

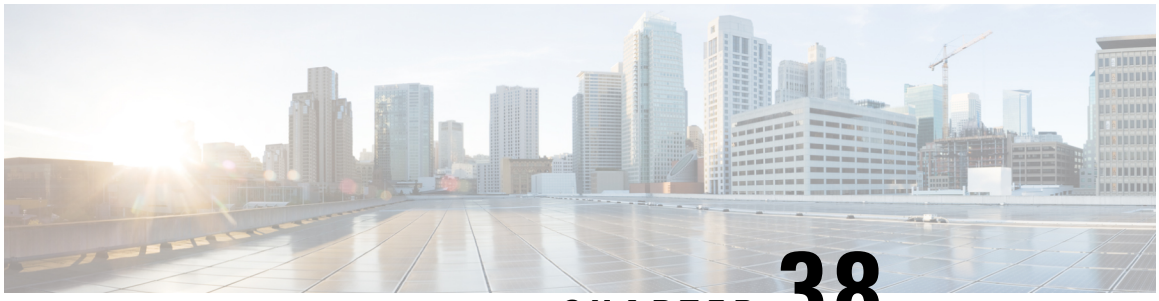
Step 5 (Optional) Delete sleeping client entries from the sleeping client cache.

- **clear wireless client sleeping-client**—Deletes all sleeping client entries from the sleeping client cache.
- **clear wireless client sleeping-client mac-address** *mac-addr*—Deletes the specific MAC entry from the sleeping client cache.

Example:

```
Device# clear wireless client sleeping-client
mac-address 00e1.e1e1.0001
```

The device displays the global configuration prompt, allowing you to make configuration changes.



CHAPTER 38

Central Web Authentication

- Central web authentication, on page 351

Central web authentication

A central web authentication is a wireless security mechanism that

- delegates the user login webpage and credential processing to a centralized authentication server such as
- initiates user redirection and authentication at Layer 2, often in combination with MAC filtering or authentication, and
- uses RADIUS server attributes so that network controllers redirect web traffic to the authentication portal. This approach streamlines access and reduces delays.

- **ISE (Identity Services Engine)**: Cisco's centralized network access policy platform that delivers authentication, authorization, accounting, and guest access functionality.
- **ACL (Access control list)**: A set of rules used to control network traffic and enforce security policies, determining which clients can access which network resources.
- **Change of Authorization (CoA)**: A mechanism to reapply or update authorization parameters dynamically without requiring a new login.

Note Currently, Cisco Identity Services Engine (ISE) supports only one Attribute-Value Pair (AVP) for Access Control Lists (ACLs), either IPv4 or IPv6. This limitation prevents the simultaneous return of both ACLs within the same authentication profile. As a result, dual-stack clients (devices using both IPv4 and IPv6) can operate within a Central Web Authentication (CWA) flow, but only under specific configurations. To avoid issues, it is recommended to configure a default IPv6 ACL with a 'deny all' rule. This ensures that clients do not encounter connectivity problems while the limitation is addressed through a future feature enhancement

Prerequisite

Comparison with other authentication methods

Central web authentication (CWA) lets you manage web-based access for wireless clients through a dedicated authentication portal (typically). Unlike traditional local web authentication, CWA performs core security and redirection at Layer 2, working with authentication mechanisms such as MAC filtering or . The RADIUS server sends specific attributes to the controller, which then redirects web traffic for portal-based login. This approach helps streamline authentication, reduce delays, and improve policy enforcement.

There are multiple methods of web authentication used in wireless environments.

- **Local web authentication (LWA)**: Local web authentication (LWA) is Layer 3 security configured on the controller. The authentication page and pre-authentication ACL are configured locally. The controller intercepts HTTP(S) traffic from the client and redirects the client to an internal authentication web page. The controller authenticates credentials locally or through RADIUS or LDAP.
- **External web authentication (EWA)**: Also configured as Layer 3 security on the controller. The controller intercepts HTTP(S) traffic and redirects the client to a login page hosted on an external web server. Credentials are authenticated by the controller locally or through a RADIUS or LDAP server. The pre-authentication ACL is statically configured on the controller.
- **Central web authentication (CWA)**: Typically configured as Layer 2 security, with the redirection URL and pre-authentication ACL residing on . During Layer 2 authentication, pushes the redirection attributes to the controller. The controller redirects all web traffic from the client to the login page. then validates the credentials entered by the client through HTTPS and authenticates the user.

This table compares different types of authentication methods used in wireless environments.

Table 16: Difference from other authentication types

Method	Layer	Portal	Credential processing	Key attributes
Local web authentication	Layer 3	Controller	Locally or through RADIUS or LDAP	Controller intercepts traffic
External web authentication	Layer 3	External server	Locally or through RADIUS or LDAP	Static ACLs and external portal
Central web authentication	Layer 2		Through (HTTPS)	

Analogy: concert tickets

Imagine a concert venue with several ways to check tickets and admit guests. You can have ticket booths at every entrance (local authentication), ticket checkers who send guests to a special desk outside (external authentication), or one main VIP booth at the heart of the venue that handles everyone’s tickets and access (central authentication). Let’s use this concert analogy to understand central web authentication and other methods.

At your concert venue, central web authentication (CWA) is what happens when, instead of letting every entrance or gate have their own ticket booth, you create one exclusive VIP booth—like —that manages all ticketing for everyone. Instead of waiting until a guest actually tries to enter through a particular door (the

way local booth might do), the venue's security starts checking guests' tickets as soon as they enter the red carpet. The VIP booth can give the gatekeepers special instructions: "If you don't recognize someone's ticket, redirect them straight to me!" This means the main ticketing process is handled efficiently and quickly by one central authority.

Let's look at all the ticketing strategies you could use at your concert:

Local Ticket Booth (Local Web Authentication, LWA): Every entrance has its own mini ticket booth and rules. Guards at the door check tickets and can ask guests for their info. Ticket validation is handled locally at each gate, sometimes via a backstage manager or external system.

External Ticket Desk (External Web Authentication, EWA): Instead of ticket checks at the gate, guests are sent to a desk outside the stadium. The desk is run by another company. The security at the entrance gates redirects guests and the validation can still interact with the backstage manager if needed. Rules for who gets through are set upfront.

VIP Central Ticket Booth (Central Web Authentication, CWA): The gates just check basic details (like guest's wristband color), and anyone who isn't recognized is sent straight to the main VIP booth () to have their ticket or credentials checked and get access granted for the whole event.

How to configure ISE

To configure ISE, follow these tasks:

1. Create an authorization profile.
2. Create an authentication rule.
3. Create an authorization rule.

Create an authorization profile

Define an authorization profile for central web authentication with required redirect attributes.

Before you begin

Ensure you have the required access and know the ACL name for redirection.

Procedure

Step 1	Click **Policy**, and click **Policy Elements**.
Step 2	Click **Results**.
Step 3	Expand **Authorization**, and click **Authorization Profiles**.
Step 4	Click **Add** to create a new authorization profile for central web authentication.
Step 5	In the **Name** field, enter a name for the profile. For example, CentralWebAuth.
Step 6	Choose **ACCESS_ACCEPT** from the **Access Type** drop-down list.
Step 7	Check the **Web Redirection (CWA, MDM, NSP, CPP)** check box, and choose **Centralized Web Auth** from the drop-down list.
Step 8	In the **ACL** field, enter the name of the ACL that defines the traffic to be redirected. For example, redirect.
Step 9	In the **Value** field, choose the default or customized values.

This field defines whether the displays the default or a custom web portal that the admin created.

Step 10 Click **Save**.

The authorization profile is created and is available for use in your central web authentication policies.

Create an authentication rule

Define an authentication rule to control network access based on specified conditions.

Use authentication rules to specify how users or devices are authenticated in your policy system. This rule determines which authentication methods and identity sources are applied when network access is requested.

Procedure

Step 1 In the **Policy** > **Authentication** page, click **Authentication**.

Step 2 Enter a name for your authentication rule. For example, MAB.

Step 3 In the If condition field, select the plus (+) icon.

Step 4 Choose **Compound condition**, and choose **Wireless_MAB**.

Step 5 Click the arrow next to **and ...** in order to expand the rule further.

Step 6 Click the + icon in the **Identity Source** field, and choose **Internal endpoints**.

Step 7 Choose **Continue** from the **If user not found** drop-down list.

A device can be authenticated even if its MAC address is not known when you select this option.

Step 8 Click **Save**.

Your new authentication rule is added to the authentication policy.

Create an authorization rule

Define and enforce policies to control access to network resources for users and devices.

Use authorization rules to allow or deny access based on user, device, or authentication attributes in .

You can configure many rules in the authorization policy, such as the *MAC not known*.

Procedure

Step 1 Click **Policy** > **Authorization**.

Step 2 In the **Rule Name** field, enter a name. For example: *Mac not known*.

Step 3 In the **Conditions** field, click the plus (+) icon.

Step 4 Choose **Compound Conditions**, and choose **Wireless_MAB**.

Step 5 From the settings icon, select **Add Attribute/Value** from the options.

Step 6 In the **Description** field, choose **Network Access** > **AuthenticationStatus** as the attribute from the drop down list.

Step 7 Choose the **Equals** operator.

Step 8 From the right hand field, choose **UnknownUser**.

Step 9 In the **Permissions** field, choose the name of the authorization profile you had created earlier.

The continues even though the user (or MAC) is not known.

Unknown users now see the login page. When they enter their credentials, the sends another authentication request. To support guest users, configure a rule to check if the user is in the guest group. For instance, if UseridentityGroup Equals Guest is used, all guests are included in this group.

Step 10 In the **Conditions** field, click the plus (+) icon.

Step 11 Choose **Compound Conditions**, and choose to create a new condition.

Place the new rule so it is evaluated prior to the *MAC not known* rule.

Step 12 From the settings icon, select **Add Attribute/Value** from the options.

Step 13 In the **Description** field, choose **Network Access** > **UseCase** as the attribute from the drop-down list.

Step 14 Choose the **Equals** operator.

Step 15 From the right hand field, choose **GuestFlow**.

Step 16 In the **Permissions** field, click the plus (+) icon to select a result for your rule.

You can choose **Standard** > **PermitAccess** option or create a custom profile to return the attributes that you like.

When the user is authorized on the login page, the triggers a Change of Authorization (COA). This process restarts Layer 2 authentication. If the user is identified as a guest, the user is authorized.

---

evaluates user or device access requests against these rules and enforces the appropriate policy. Unknown users see the login page and, if identified as guests, receive guest access upon authentication.

How to configure central web authentication on the controller

To configure central web authentication on the controller, proceed with these tasks.

1. Configure WLAN.
2. Configure policy profile.
3. Configure redirect ACL.
4. Configure AAA for central web authentication.
5. Configure redirect ACL in Flex profile.

Configure WLAN (GUI)

Set up a new WLAN on your wireless controller using the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 In the **WLANs** window, click the name of the **WLAN** or click **Add** to create a new one.
The **Add/Edit WLAN** window is displayed.

Step 3 In the **Add/Edit WLAN** window, click the **General** tab to configure these parameters.

- In the **Profile Name** field, enter or edit the name of the profile.
- In the **SSID** field, enter or edit the SSID name.

  The SSID name is alphanumeric and can be up to 32 characters in length.
- In the **WLAN ID** field, enter or edit the ID number. The valid range is between one and 512.
- Select the **802.11** radio band from the **Radio Policy** drop-down list.
- Using the **Broadcast SSID** toggle button, change the status to either **Enabled** or **Disabled**.
- Using the **Status** toggle button, change the status to either **Enabled** or **Disabled**.

Step 4 Click the **Security** tab, and then select the **Layer 2** tab to configure these parameters:

- Select **None** from the **Layer 2 Security Mode** drop-down list. This setting disables Layer 2 security.
- Enter the **Reassociation Timeout** value, in seconds. This value specifies the duration before a fast transition reassociation times out.
- Check the **Over the DS** check box to enable Fast Transition over a distributed system.
- Choose **OWE**. Opportunistic Wireless Encryption (OWE) provides data confidentiality with encryption over the air between an AP radio and a wireless client. OWE Transition Mode ensures backwards compatibility.
- Choose Fast Transition (802.11r), the IEEE standard for fast roaming. This standard allows the initial handshake with a new AP to occur before the client roams to the target AP. This method is known as Fast Transition.
- Check the check box to enable MAC filtering in the WLAN.

Step 5 Click **Save & Apply to Device**.

Configure WLAN (CLI)

Set up a new WLAN on your wireless controller using the CLI.

Before you begin

Enable MAC filtering for Layer 2 authentication to download the redirect URL and ACL.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN.

Example:

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

Step 3 Enable MAC filtering on a WLAN.

Example:

```
Device(config-wlan)# mac-filtering name
```

If the authentication list is not already configured, the default authentication list is used when configuring MAC filtering.

Step 4 Disable WPA security.

Example:

```
Device(config-wlan)# no security wpa
```

Step 5 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 6 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

The new WLAN is available and active on the controller.

After completing the WLAN configuration, if the changes are not pushed to all the APs, a syslog message appears.

```
2021/01/06 16:20:00.597927186 {wncd_x_R0-4}{1}: [wlanmgr-db] [20583]: UUID: 0, ra: 0, TID:
 0
```

```
(note): Unable to push WLAN config changes to all APs, cleanup required for WlanId: 2,
profile: wlan1 state: Delete pending
```

If this syslog message appears for more than six minutes, reload the controller

If the controller does not reload and the syslog message continues to appear, collect the archive logs and wncd core file. Then, raise a case by clicking the link: Support Case Manager.

```
Device# config terminal
            Device(config)# wlan wlanProfileName 1 ngwcSSID
            Device(config-wlan)# mac-filtering default
            Device(config-wlan)# no security wpa
            Device(config-wlan)# no shutdown
            Device(config-wlan)# end
```

Configure policy profile (CLI)

Define and activate a policy profile to control WLAN behavior using the CLI.

Use the CLI to set up policy profiles that govern AAA, NAC, and VLAN assignment for wireless clients on Cisco devices. Apply this procedure when building new WLAN policies or adjusting network admission controls.

Before you begin

- Both NAC and AAA override must be configured in the policy profile for proper CWA operation.
- You need a AAA override to apply policies coming from the AAA or ISE servers. When a redirect URL and redirect ACL is received from the ISE server, NAC is used to trigger the Central Web Authentication (CWA).
- The default policy profile is used for APs not associated with a specific policy.
- Clients may experience VLAN assignment during the final RADIUS access-accept following successful authentication.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN.

Example:

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

Step 3 Set the policy profile.

Example:

```
Device(config)# wireless profile policy default-policy-profile
```

Step 4 Map the VLAN to a policy profile.

Example:

```
Device(config-wireless-policy)# vlan vlan-id
```

If vlan-id is not specified, the default native vlan one is applied. The valid range for vlan-id is one to 4096.

Management VLAN is applied if no VLAN is configured on the policy profile.

Step 5 Configure AAA override to apply policies coming from the AAA or ISE servers.

Example:

```
Device(config-wireless-policy)# aaa-override
```

Step 6 Configure Network Access Control in the policy profile. NAC is used to trigger the Central Web Authentication (CWA).

Example:

```
Device(config-wireless-policy)# nac
```

Step 7 Enable the WLAN.

Example:

```
Device(config-wireless-policy)# no shutdown
```

Step 8 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

The policy profile is configured and active, applying the desired AAA, NAC, and VLAN parameters to WLAN clients.

```
Device# configure terminal
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# vlan 41
Device(config-wireless-policy)# aaa-override
Device(config-wireless-policy)# nac
Device(config-wireless-policy)# no shutdown
Device(config-wireless-policy)# end
```

Configure a policy profile (GUI)

Create and apply a new policy profile using the GUI.

Use this task to define access and traffic management settings by configuring a policy profile in the system's GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click **Add**.

Step 3 In the **Add Policy Profile** window, in General Tab, enter a name and description for the policy profile.

Step 4 To enable the policy profile, set **Status** as Enabled.

Step 5 Use the slider to enable or disable **Passive Client** and **Encrypted Traffic Analytics** .

Step 6 (Optional) In the **CTS Policy** section, choose the appropriate status for the following:

- Inline Tagging—a transport mechanism using which a , controller, or AP understands the source SGT.
- SGACL Enforcement

Step 7 Specify a default **SGT**. The valid range is from two to 65519.

Step 8 In the **WLAN Switching Policy** section, choose as required:

- Central Switching
- Central Authentication
- Central DHCP
- Central Association Enable
- Flex NAT/PAT

Step 9 Click **Save & Apply to Device**.

The new policy profile is created and applied to the device using your configured settings.

Create redirect ACL

Create a redirect access control list (ACL). Direct unauthenticated HTTP and HTTPS traffic to the login page. Allow traffic to and block other traffic

The redirect ACL is a punt ACL that must be predefined on the controller. For FlexConnect local switching, the ACL must be predefined on the AP. The Authentication, authorization, and accounting (AAA) server server returns only the name of the ACL, not its definition.

The redirect ACL defines traffic matching "deny" statements, allowing this traffic to pass through the data plane. Traffic matching "permit" statements is sent to the control plane for further processing, which includes web interception and redirection.

The redirect ACL includes implicit statements that allow DHCP and DNS traffic to all IP addresses, similar to LWA.

It also ends with an implicit deny statement in the security ACL.

Before you begin

Ensure you have the IP address of your deployment.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create an extended access list.

Example:

```
Device(config)# ip access-list extended redirect
```

If users are not authenticated, HTTP and HTTPS browsing fails because uses a redirect ACL (named **redirect**).

Step 3 Permit traffic to and block all other traffic.

Example:

```
Device(config)# deny ip any host ISE-IP-address
```

Step 4 Permit traffic to and all other traffic is blocked.

Example:

```
Device(config)# deny ip host ISE-IP-add any
```

Note
This ACL is applicable for both local and flex mode.

Step 5 Redirect all HTTP or HTTPS access to the ISE login page. Port number 80 is used for HTTP, and port number 443 is used for HTTPS, configured with the **permit TCP any any eq web address/port-number** command.

Example:

In case of HTTP:

```
Device(config)# permit TCP any any eq www
Device(config)# permit TCP any any eq 80
```

Example:

In case of HTTPS:

```
Device(config)# permit TCP any any eq 443
```

To allow traffic to , configure before the HTTP or HTTPS ACE entry.

Step 6 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

HTTP

```
Device# configure terminal
Device(config)# ip access-list extended redirect
Device(config)# deny ip any host 123.123.134.112
Device(config)# deny ip host 123.123.134.112 any
Device(config)# permit TCP any any eq www
Device(config)# permit TCP any any eq 80
Device(config)# end
```

HTTPS

```
Device# configure terminal
Device(config)# ip access-list extended redirect
Device(config)# deny ip any host 123.123.134.112
Device(config)# deny ip host 123.123.134.112 any
Device(config)# permit TCP any any eq 443

Device(config)# end
```

Configure AAA for CWA

Set up AAA to support Central Web Authentication using CLI.

Central Web Authentication requires proper AAA configuration to enable Change of Authorization (CoA) functions using a RADIUS (Remote Authentication Dial-In User Service) server.

Before you begin

Ensure AAA authentication is configured on your device. For more details, see the *Configuring AAA Authentication* documentation.

Procedure

Step 1 Configure the Change of Authorization (CoA) on the .

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 2 Specify a RADIUS client and configure the RADIUS key that is shared between the device and the RADIUS client.

Example:

```
Device(config-locsvr-da-radius)# client ISE-IP-add server-key
radius-shared-secret
```

ISE-IP-add is the IP address of the RADIUS client.

server-key is the radius client server-key.

radius-shared-secret can have these values:

- **0**—Specifies unencrypted key.
- **6**—Specifies encrypted key.
- **7**—Specifies HIDDEN key.
- **Word**—Unencrypted (cleartext) server key.

When configuring WSMA data in GUI, ensure that the RADIUS shared secret does not exceed 240 characters.

AAA is configured to support Central Web Authentication with dynamic CoA, enabling secure device communication with the RADIUS server.

```
Device# config terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 123.123.134.112 server-key 0 SECRET
Device(config-locsvr-da-radius)# end
```

Configure redirect ACL in flex profile (GUI)

Set up a redirect ACL within a flex profile so that APs enforce specified access control and web authentication policies.

The redirect ACL definition must be sent to the AP in the FlexConnect profile. The redirect ACL associated with an AP must be configured in the FlexConnect profile that hosts the client. If an AP is not configured with a FlexConnect profile, it is associated with the default FlexConnect profile.

Before you begin

- Identify the FlexConnect profile to be updated or verify if a new Flex profile needs to be created.
- Prepare the ACL details, central web authentication requirements, and any preauthentication URL filters.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Flex**.

Step 2 On the **Flex Profile** page, click the name of the FlexConnect profile or click **Add** to create a new FlexConnect profile.

Step 3 In the **Add/Edit Flex Profile** window that is displayed, click the **Policy ACL** tab.

Step 4 Click **Add** to map an ACL to the FlexConnect profile.

Step 5 Choose the ACL name, enable central web authentication, and specify the preauthentication URL filter.

Step 6 Click **Save**.

Step 7 Click **Update & Apply to Device**.

The AP applies the configured redirect ACL parameters for client traffic as specified in the flex profile.

Configure redirect ACL in flex profile (CLI)

Set up a redirect ACL within a flex profile so that APs enforce specified access control and web authentication policies.

The redirect ACL definition must be sent to the AP in the FlexConnect profile. The redirect ACL associated with an AP must be configured in the FlexConnect profile that hosts the client. If an AP is not configured with a FlexConnect profile, it is associated with the default FlexConnect profile.

Note When the ACL is pushed down to the APs, the permission must change from **deny** to **permit** or the other way around. This change does not occur if the ACL contains an object group, causing the ACL not to be fully translated, which may cause the redirection to fail.

Before you begin

- Ensure you have the required ACL policy created on the controller.
- Confirm that AP is operating in FlexConnect mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new flex policy.

Example:

```
Device(config)# wireless profile flex default-flex-profile
```

The default flex profile name is **default-flex-profile**.

Step 3 Configure an ACL policy.

Example:

```
Device(config-wireless-flex-profile)# acl-policy
                        acl-policyname
```

Step 4 Configure central web authentication.

Example:

```
Device(config-wireless-flex-profile-acl)# central-webauth
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-wireless-flex-profile-acl)# end
```

The AP in flex profile has the redirect ACL configured. The ACL is now applied and used for client redirection during centralized web authentication.

Troubleshoot CWA

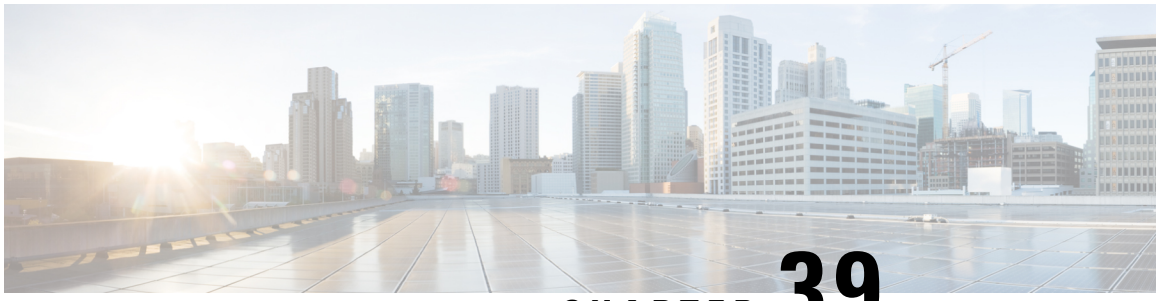
Init-State timer running out

Problem Issue: The client devices are deauthenticated by the controller if users fail to enter their credentials in a limited time interval. The clients are deauthenticated after three times the time configured for the init-state timeout in the controller.

Problem Explanation: This is expected behavior. For central web authentication, the init-state timeout does not apply directly. The reap timer—three times the init-state timeout plus five seconds (3 × init-state timeout + 5)—determines when your device deauthenticates in seconds.

Problem For example, if you have configured the init-state timeout as 10 seconds, then the client devices are deauthenticated if users fail to enter their credentials after 35 seconds; that is (3*10 + 5) = 35 seconds.

Problem For example, if you set the init-state timeout to 10 seconds, your device deuathenticates after 35 seconds if you do not enter your credentials. (This calculation is: 3 × 10 + 5 = 35 seconds.)



CHAPTER 39

ISE Simplification and Enhancements

- Security utilities configuration options, on page 367
- Captive bypassing, on page 370
- DHCP option 55 and 77 parameters, on page 372
- Captive portals, on page 376

Security utilities configuration options

You can set up the RADIUS server using this simplified method:

wireless-default radius server *ip* **key** *secret*

- Configures AAA authorization for network services, including authentication for web authentication and Dot1x.
- Enables local authentication with default authorization.
- Applies the default redirect ACL for Central Web Authentication (CWA).
- Creates a global parameter map with a virtual IP and enables captive bypass portal.
- Configures all default AAA settings required when configuring a RADIUS server.
- Sets method-list configuration and enables RADIUS accounting by default.
- Disables RADIUS aggressive failovers by default and sets the RADIUS request timeout to five seconds.
- Implements a default access control list (ACL) for CWA URL redirect.

```
aaa new-model
               aaa authentication webauth default group radius
               aaa authentication dot1x default group radius
               aaa authorization network default group radius
               aaa accounting identity default start-stop group radius
               !
               aaa server radius dynamic-author
               client <IP> server-key cisco123
               !
               radius server RAD_SRV_DEF_<IP>
               description Configured by wireless-default
               address ipv4 <IP> auth-port 1812 acct-port 1813
               key <key>
               !
```

```
aaa local authentication default authorization default
aaa session-id common
!
ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
remark " CWA ACL to be referenced from ISE "
deny udp any any eq domain
deny tcp any any eq domain
deny udp any eq bootps any
deny udp any any eq bootpc
deny udp any eq bootpc any
deny ip any host <IP>
permit tcp any any eq www
!
parameter-map type webauth global
captive-bypass-portal
virtual-ip ipv4 192.0.2.1
virtual-ip ipv6 2001:DB8::1
!
wireless profile policy default-policy-profile
aaa-override
local-http-profiling
local-dhcp-profiling
accounting
```

Key benefits

- Simplifies deployment of RADIUS server configuration for wireless controllers.
- Eliminates the need for step-by-step manual configuration across multiple CLI commands.
- Ensures consistent and secure default settings for AAA and authentication services.

Configure multiple RADIUS servers

To configure multiple RADIUS servers for authentication in a network.

Before you begin

Ensure you have the necessary IP addresses and shared secrets for the RADIUS servers.

Procedure

Step 1 Enter global configuration mode

Example:

```
Device# configure terminal
```

Step 2 Configure a RADIUS server

Example:

```
Device(config)# wireless-default radius server ip key secret
```

Example:

```
Device(config)# wireless-default radius server 192.2.58.90 key cisco123
```

You can configure up to ten RADIUS servers.

Step 3 Return to privileged EXEC mode

Example:

```
Device(config)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Multiple RADIUS servers are now configured and ready for use in the network.

Verify AAA and Radius server configurations

To view details of AAA server, use these command:

```
Device# show run aaa
!
aaa new-model
aaa authentication webauth default group radius
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa accounting Identity default start-stop group radius
!
aaa server radius dynamic-author
 client 192.0.2.10 server-key cisco123
!
radius server RAD_SRV_DEF_192.0.2.10
 description Configured by wireless-default
 address ipv4 192.0.2.10 auth-port 1812 acct-port 1813
 key cisco123
!
aaa local authentication default authorization default
aaa session-id common
!
!
ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
remark " CWA ACL to be referenced from ISE "
deny udp any any eq domain
deny tcp any any eq domain
deny udp any eq bootps any
deny udp any any eq bootpc
deny udp any eq bootpc any
deny ip any host 192.0.2.10
permit tcp any any eq www
!
parameter-map type webauth global
  captive-bypass-portal
  virtual-ip ipv4 192.0.2.1
  virtual-ip ipv6 2001:DB8::1
!
wireless profile policy default-policy-profile
   aaa-override
   local-http-profiling
   local-dhcp-profiling
   accounting
```

Note The **show run aaa** output may change when new commands are added to this utility.

Captive bypassing

A captive bypass is a network behavior that

- enables client devices to automatically determine Internet connectivity and initiate authentication flows,
- may enable users to bypass captive portal splash pages through protocols such as WISPr.
- involves interaction between devices (such as Apple iOS) and Embedded Wireless Controllers, affecting web redirection and authentication.

Additional reference information

WISPr is a draft protocol that enables users to roam between different wireless service providers. Devices like Apple iOS use HTTP WISPr requests to check if they are connected to the Internet.

When direct Internet connectivity is not detected, these mechanisms prompt the device to open a web browser so the user can enter credentials for Internet access. Authentication occurs in the background whenever the device connects to a new SSID.

When an Apple iOS device connects, it sends a WISPr request to the Embedded Wireless Controller, which checks the user agent details and then triggers an HTTP request with web authentication interception. After verifying the iOS version and browser details, the Embedded Wireless Controller allows the client device to bypass captive portal settings and access the Internet.

This HTTP request triggers web authentication interception on the Embedded Wireless Controller just as any other page requests do. This leads to the standard web authentication process. If web authentication uses splash page features (such as a URL from a configured RADIUS server), the splash page might not appear for the user because WISPr requests occur at short intervals. Once a query reaches the server, web redirection or splash page display running in the background is cancelled, and the device processes the page request, which may prevent the splash page from being displayed as intended.

Example

For instance, Apple introduced an iOS feature that detects captive portals by sending a web request upon connecting to a wireless network. On iOS 6 and earlier, the request is sent to http://www.apple.com/library/test/success.html. On iOS 7 and later, multiple possible URLs are used. If a response is received, Internet access is assumed and no further interaction occurs. If no response is received, Internet access is assumed to be blocked and the device's Captive Network Assistant (CNA) auto-launches a pseudo-browser to prompt for portal login. The CNA may not function properly if it is redirected to an ISE captive portal. The Embedded Wireless Controller can suppress the pop-up of the pseudo-browser. Network administrators can configure the Embedded Wireless Controller to bypass WISPr detection. In this configuration, web authentication interception only occurs when a user intentionally requests a web page, ensuring the splash page is loaded in the user's browser context and not triggered by background connectivity probes.

Analogy: Sneaking into an amusement park

Captive bypassing is like someone sneaking into an amusement park through a side door, instead of showing their ticket at the main entrance. Some devices use automated checks to find shortcuts and access the internet without going through the intended login page, just as clever visitors might bypass security by finding an

easier way in. Network controls are needed to make sure everyone enters through the main gate and checks in properly.

Configure captive bypassing for WLAN in LWA and CWA (GUI)

Enable client devices to bypass captive portal authentication on the selected WLAN.

Before you begin

Use this task when you need certain clients to connect without passing through the captive portal, typically for pre-authorized devices.

Ensure you have administrative access to the device and the parameter map is appropriately configured.

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth** .

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 Select **Captive Bypass Portal** check box.

Step 4 Click **Update & Apply to Device** .

The selected WLAN now allows captive bypass in accordance with your configuration.

Configure captive bypassing for WLAN in LWA and CWA

Enable captive bypassing for WLAN to allow users to bypass the captive portal.

This configuration is applicable in environments where users need to access the network without going through a captive portal.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create the parameter map.

Example:

```
Device(config)# parameter-map type webauth parameter-map-name
```

The *parameter-map-name* must not exceed 99 characters.

Step 3 Configure captive bypassing.

Example:

```
Device(config)# captive-bypass-portal
```

Step 4 Specify the WLAN name and ID.

Example:

```
Device(config)# wlan profile-name wlan-id ssid-name
```

- *profile-name* is the WLAN name which can contain 32 alphanumeric characters.
- *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 512.
- *ssid-name* is the SSID which can contain 32 alphanumeric characters.

Step 5 Enable the web authentication for the WLAN.

Example:

```
Device(config-wlan)# security web-auth
```

Step 6 Map the parameter map.

Example:

```
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

If the parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

Step 7 (Optional) Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The WLAN is now configured with captive bypassing enabled, allowing users to bypass the captive portal.

DHCP option 55 and 77 parameters

The DHCP sensors use these DHCP options on the ISE for both native and remote device profiling:

- Option 12: Hostname
- Option 6: Class Identifier

These options must also be sent to ISE for accurate profiling:

- Option 55: Parameter Request List
- Option 77: User Class

Configure the DHCP options 55 and 77 to ISE using the GUI

Ensure DHCP options 55 and 77 are sent to ISE to enable device profiling and policy enforcement

These options allow ISE to collect device-specific DHCP information for accurate profiling on wireless networks.

Before you begin

Confirm that your network devices are compatible with ISE profiling features.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click **Add** to view the **Add Policy Profile** window.

Step 3 Click **Access Policies** tab, choose the **RADIUS Profiling** and **DHCP TLV Caching** check boxes to configure radius profiling and DHCP TLV Caching on a WLAN.

Step 4 Click **Save & Apply to Device**

he network device now sends DHCP options 55 and 77 to ISE, enabling accurate device profiling and network access policy enforcement.

Configure the device to send DHCP options 55 and 77 to ISE (CLI)

Enable the device to send specific DHCP options to ISE for improved client profiling.

This configuration is essential for environments where client profiling is necessary for network access control and monitoring.

Before you begin

Ensure that the device is running a compatible version of the software that supports DHCP options 55 and 77.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure WLAN policy profile and enters the wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy profile-policy
```

The *profile-policy* is the name of the WLAN policy profile being configured.

Step 3 Configure DHCP TLV caching on a WLAN.

Example:

```
Device(config-wireless-policy)# dhcp-tlv-caching
```

Enables caching of DHCP TLV information for better profiling.

Step 4 Configure client radius profiling on a WLAN.

Example:

```
Device(config-wireless-policy)# radius-profiling
```

Enables RADIUS profiling to enhance client identification.

Step 5 (Optional) Return to privileged EXEC mode.

Example:

```
Device(config-wireless-policy)# end
```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The device is configured to include DHCP options 55 and 77 in client profiling information sent to Cisco ISE.

Configure the EAP request timeout using the GUI

Use the steps to configure the EAP request timeout through the GUI:

Procedure

Step 1 Choose **Configuration** > **Security** > **Advanced EAP**.

Step 2 In the **EAP-Identity-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP identity request to wireless clients using local EAP.

Step 3 In the **EAP-Identity-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP identity request to wireless clients using local EAP.

Step 4 Set **EAP Max-Login Ignore Identity Response** to **Enabled** state to limit the number of clients that can be connected to the device with the same username. You can log in up to eight times from different clients (PDA, laptop, IP phone, and so on) on the same device. The default state is **Disabled**.

Step 5 In the **EAP-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP request to wireless clients using local EAP.

Step 6 In the **EAP-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP request to wireless clients using local EAP.

Step 7 In the **EAPOL-Key Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP key over the LAN to wireless clients using local EAP.

Step 8 In the **EAPOL-Key Max Retries** field, specify the maximum number of times that the device attempts to send an EAP key over the LAN to wireless clients using local EAP.

Step 9 In the **EAP-Broadcast Key Interval** field, specify the time interval between rotations of the broadcast encryption key used for clients and click **Apply**.

Note
After configuring the EAP-Broadcast key interval to a new time period, you must shut down or restart the WLAN for the changes to take effect. Once the WLAN is shut down or restarted, the M5 and M6 packets are exchanged when the configured timer value expires.

Configure EAP request timeout

Set the timeout for EAP requests to manage client exclusion effectively.

This configuration is useful in environments where EAP authentication is used, and it is necessary to handle clients that do not respond in a timely manner.

Before you begin

Ensure you have access to the device and are in the appropriate configuration mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable exclusion on timeout and no response.

Example:

```
Device(config)# wireless wps client-exclusion dot1x-timeout
```

By default, this feature is enabled. To disable, append a *no* at the beginning of the command.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The EAP request timeout is now configured, allowing for better management of client exclusions.

Configure the EAP request timeout in wireless security

Set the EAP request timeout to ensure timely authentication in wireless security.

This configuration is essential in environments where EAP is used for authentication, as it affects the responsiveness of the authentication process.

Before you begin

Ensure you have access to the device and are in privileged EXEC mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the EAP request retransmission timeout value in seconds.

Example:

```
Device(config)# wireless security dot1x request timeout timeout
```

Configures the EAP request retransmission timeout value in seconds.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The EAP request timeout is now configured, which will help in managing the authentication process in wireless security.

Captive portals

A captive portal is a network authentication mechanism that

- intercepts and redirects user web requests to a designated login or acceptance page,
- allows administrators to enforce authentication or policy acceptance before granting network access, and
- supports configuration of multiple web authentication URLs based on access point (AP), WLAN, or global scope.

Order of precedence

Captive portal configurations let you define different web authentication URLs—including external URLs—for the same SSID, based on the specific access point or WLAN. By default, the system uses a global URL for user authentication, but administrators may override this at the WLAN or AP level to customize user experience.

Order of precedence

1. AP-level configuration (highest priority)
2. WLAN-level configuration
3. Global configuration (default)

Restrictions

- This feature is supported on standalone controllers only.
- Export-Anchor configurations are not supported.

Configure captive portal (GUI)

Configure a secure WLAN that uses captive portal authentication for guest or user access.

Configure a captive portal when you require user authentication on your Wi-Fi network for guests or employees.

Before you begin

Confirm the web authentication parameter map and necessary authentication lists are already created.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.

Step 4 In the **Security** > **Layer2** tab, uncheck the **WPA Policy**, **AES** and **802.1x** check boxes.

Step 5 In the **Security** > **Layer3** tab, choose the parameter map from the **Web Auth Parameter Map** drop-down list and authentication list from the **Authentication List** drop-down list.

Step 6 In the **Security** > **AAA** tab, choose the Authentication list from the **Authentication List** drop-down list.

Step 7 Click **Apply to Device**.

Step 8 Choose **Configuration** > **Security** > **Web Auth**.

Step 9 Choose a **Web Auth Parameter Map**.

Step 10 In the **General** tab, enter the **Maximum HTTP connections**, **Init-State Timeout(secs)** and choose **webauth** from the **Type** drop-down list.

Step 11 In the **Advanced** tab, under the **Redirect to external server** settings, enter the **Redirect for log-in** server.

Step 12 Click **Update & Apply**.

The captive portal WLAN is available, and users associating with this SSID will be redirected for authentication as defined in the parameter map.

Configure captive portal

Set up a captive portal to manage user access to the network.

The captive portal is used to authenticate users before granting them access to the network, typically in public Wi-Fi environments.

Before you begin

Ensure that the WLAN is properly configured and that you have the necessary access rights.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the WLAN profile.

Example:

```
Device(config)# wlan profile-name
```

Enables or disables all WLANs and creates the WLAN identifier. The *profile-name* and the SSID network name should be up to 32 alphanumeric characters.

Step 3 Configure the WLAN web ACL.

Example:

```
Device(config-wlan)# ip access-group web IPv4-ACL-Name
```

WLAN needs to be disabled before performing this operation.

Step 4 Disable WPA security.

Example:

```
Device(config-wlan)# no security wpa
```

Step 5 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Disables security AKM for dot1x.

Step 6 Disable WPA2 ciphers for AES.

Example:

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

Disables WPA2 ciphers for AES.

Step 7 Enable web authentication for WLAN.

Example:

```
Device(config-wlan)# security web-auth authentication-list authentication-list-name
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

Enables web authentication for WLAN. Here,

- **authentication-list**

  authentication-list-name : Sets the authentication list for IEEE 802.1x.

- **authorization-list**

  authorization-list-name : Sets the override-authorization list for IEEE 802.1x.

- **on-macfilter-failure** : Enables Web authentication on MAC filter failure.

- **parameter-map**

  parameter-map-name : Configures the parameter map.

Note
When **security web-auth** is enabled, you get to map the default **authentication-list** and global **parameter-map** . This is applicable for authentication-list and parameter-map that are not explicitly mentioned.

Step 8 (Optional) Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 9 Exit from the WLAN configuration.

Example:

```
Device(config-wlan)# exit
```

Step 10 Create a parameter map and enter parameter-map webauth configuration mode.

Example:

```
Device(config)# parameter-map type webauth parameter-map-name
```

Creates a parameter map and enters parameter-map webauth configuration mode.

Step 11 Configure the webauth type parameter.

Example:

```
Device(config-params-parameter-map)# type webauth parameter-map-name
```

Step 12 Configure the WEBAUTH timeout in seconds.

Example:

```
Device(config-params-parameter-map)# timeout init-state sec <timeout-seconds>
```

Valid range for the time in sec parameter is 60 seconds to 3932100 seconds.

Step 13 Configure the URL string for redirect during login.

Example:

```
Device(config-params-parameter-map)# redirect for-login <URL-String>
```

Example:

```
Device(config-params-parameter-map)# redirect for-login
https://172.16.100.157/portal/login.html
```

Configures the URL string for redirect during login.

Step 14 Exit the parameters configuration.

Example:

```
Device(config-params-parameter-map)# exit
```

Step 15 Configure policy tag and enter policy tag configuration mode.

Example:

```
Device(config)# wireless tag policy policy-tag-name
```

Step 16 Attach a policy profile to a WLAN profile.

Example:

```
Device(config-policy-tag)# wlan wlan-profile-name policy policy-profile-name
```

Step 17 Save the configuration and exit configuration mode.

Example:

```
Device(config-policy-tag)# end
```

The captive portal is now configured and ready to manage user access to the network.

Example: Captive portal configuration

The example shows how you can have APs at different locations, broadcasting the same SSID but redirecting clients to different redirect portals:

Configuring multiple parameter maps pointing to different redirect portal:

```
parameter-map type webauth parMap1
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.3:8080/portal/PortalSetup.action?portal=cfdbce00-2ce2-11e8-b83c-005056a06b27
redirect portal ipv4 172.16.12.3
!
!
parameter-map type webauth parMap11
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.4:8443/portal/PortalSetup.action?portal=094e7270-3808-11e8-9797-02421e4cae0c
redirect portal ipv4 172.16.12.4
!
```

Associating these parameter maps to different WLANs:

```
wlan edc1 1 edc
ip access-group web CPWebauth
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap11
no shutdown
wlan edc2 2 edc
ip access-group web CPWebauth
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap1
no shutdown
```



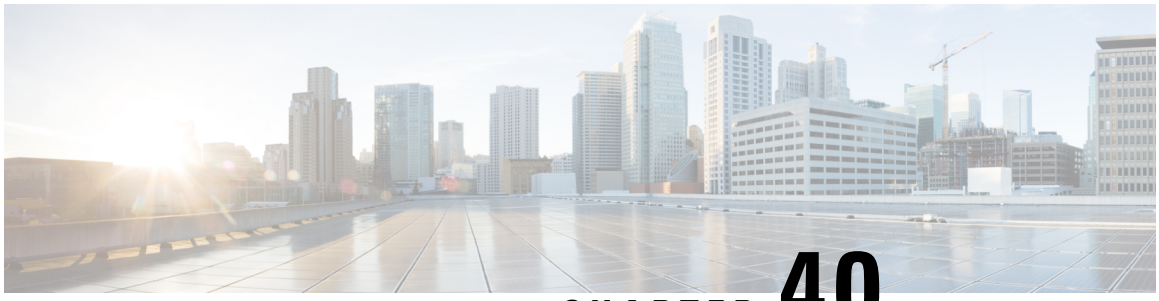
Note All WLANs have identical SSIDs.

Associating WLANs to different policy tags:

```
wireless tag policy policy_tag_edc1
wlan edc1 policy policy_profile_flex
wireless tag policy policy_tag_edc2
wlan edc2 policy policy_profile_flex
```

Assigning these policy tags to the desired APs:

```
ap E4AA.5D13.14DC
policy-tag policy_tag_edc1
site-tag site_tag_flex
ap E4AA.5D2C.3CAC
policy-tag policy_tag_edc2
site-tag site_tag_flex
```



CHAPTER 40

Authentication and Authorization Between Multiple RADIUS Servers

- Authentication and authorization between multiple RADIUS servers, on page 381
- Configure web authentication for WLAN with split authentication and authorization servers, on page 388

Authentication and authorization between multiple RADIUS servers

A RADIUS authentication and authorization role is a function within a RADIUS-based wireless controller configuration that:

- allows authentication and authorization duties to be carried out by a single RADIUS server or split between different servers
- enables the controller to relay authentication attributes to a separate authorization server, and
- in a split configuration, requires both servers to return ACCESS-ACCEPT for a client session to be established.

Cisco Embedded Wireless Controller on Catalyst Access Points use a request and response transaction with a single RADIUS server that combines both authentication and authorization.

A RADIUS server can assume the role of either an authentication server, authorization server, or both. In cases where there are separate RADIUS servers for authentication and authorization, the Session Aware Networking (SANet) component on the embedded wireless controller allows authentication on one server and authorization on another when a client joins embedded wireless controller.

Authentication can be performed using Cisco ISE, Cisco Catalyst Center, Free RADIUS, or any third-party RADIUS Server. After the authentication server successfully authenticates a user, the embedded wireless controller relays attributes received from the authentication server to another RADIUS server designated as the authorization server.

The authorization server then performs these:

- Processes received attributes with the other policies or rules defined on the server.

- Derives attributes as part of the authorization response and returns them to the embedded wireless controller.

Note The Authentication/Authorization list created through Cisco Catalyst Center provisioning supports a maximum of 100 entries. Entries beyond 100 can be created, but they do not function.

Configure explicit authentication and authorization server list (GUI)

Set up and customize authentication and authorization server lists to control network access and user verification methods using the GUI.

Designate specific authentication and authorization servers, such as RADIUS, TACACS+, or LDAP, to handle AAA (Authentication, Authorization, and Accounting) operations on your device.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**. On the **Authentication Authorization and Accounting** page, click the **Servers/Groups** tab. Click the type of AAA server you want to configure from these options:

- RADIUS
- TACACS+
- LDAP

In this procedure, the RADIUS server configuration is described.

Step 2 With the **RADIUS** option selected, click **Add** and enter a name for the RADIUS server and the IPv4 or IPV6 address of the server.

Step 3 Enter the authentication and encryption key to be used between the device and the key string RADIUS daemon running on the RADIUS server. You can choose to use either a PAC key or a non-PAC key.

Step 4 Enter the server timeout value; the valid range is 1 to 1000 seconds. Enter a retry count; the valid range is 0 to 100.

Step 5 Leave the **Support for CoA** field in **Enabled** state and click **Save & Apply to Device**.

Step 6 On the **Authentication Authorization and Accounting** page, with **RADIUS** option selected, click the **Server Groups** tab. In the **Create AAA RADIUS Server Group** window that is displayed, enter a name for the RADIUS server group.

Step 7 From the **MAC-Delimiter** drop-down list, select the delimiter to be used in the MAC addresses that are sent to the RADIUS servers. From the **MAC Filtering** drop-down list, select a value based on which to filter MAC addresses.

Step 8 To configure dead time for the server group and direct AAA traffic to alternative groups of servers that have different operational characteristics, in the **Dead-Time** field, enter the amount of time, in minutes, after which a server is assumed to be dead.

Step 9 Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list. Click **Save & Apply to Device**.

Configure explicit authentication server list (GUI)

Set up and manage RADIUS authentication servers and server groups to control authentication and accounting within your network environment using the GUI.

Use this task when you need to explicitly designate which RADIUS servers and groups are available for authentication in your AAA configuration, using the device's graphical interface.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.

Step 2 Choose **RADIUS** > **Servers** tab.

Step 3 Click **Add** to add a new server or click an existing server.

Step 4 Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port**. Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**.

Step 5 Click **Apply to Device**.

Step 6 Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.

Step 7 Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.

Step 8 Click **Apply to Device**.

Configure explicit authentication server list (CLI)

Set up an authentication server list by explicitly defining RADIUS server parameters using commands.

Use this procedure on Cisco devices when you require explicit configuration of authentication server parameters for AAA.

Procedure

Step 1 Enable the privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server name
```

Step 4 Specify the RADIUS server parameters.

Example:

```
Device(config-radius-server)# address ipv4 192.0.2.56 auth-port 1812 acct-port 1813
```

Step 5 Specify the authentication and encryption key used between the device and the key string RADIUS daemon running on the RADIUS server.

Example:

```
Device(config-radius-server)# pac key key
```

Step 6 Return to the configuration mode.

Example:

```
Device(config-radius-server)# exit
```

Step 7 Create a radius server-group identification.

Example:

```
Device(config)# aaa group server radius server-group
```

server-group refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

If the IP address of the RADIUS server is not added to the routes defined for the controller, the default route is used. We recommend that you define a specific route to source the traffic from the defined SVI in the AAA server group.

Step 8 Configure the server name.

Example:

```
Device(config)# server name server-name
```

Step 9 Return to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example:

```
Device(config)# end
```

For more information, see *Configuring AAA for External Authentication*.

Configure explicit authorization server list (GUI)

Set up and organize RADIUS authorization servers and groups for secure network authentication and accounting using the GUI.

Configuring an explicit authorization server list ensures that devices can authenticate users and account activity through specified RADIUS servers. This centralized management helps enforce access policies and simplifies administration.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.

Step 2 Choose **RADIUS** > **Servers** tab.

Step 3 Click **Add** to add a new server or click an existing server.

Step 4 Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port.** Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**.

Step 5 Click **Apply to Device**.

Step 6 Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.

Step 7 Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.

Step 8 Click **Apply to Device**.

Configure explicit authorization server list (CLI)

You must direct AAA validation access requests to a specific server. Otherwise, the wireless controller selects any available AAA server from the configured pool to validate access requests. Because the controller cannot target a specific server for validation, you may face inefficiencies and potential security vulnerabilities.

To address this, an authorization method list is created to allow administrators to explicitly direct the controller to use a specific AAA server or the local database for validation. This configuration ensures a streamlined and secure validation process by eliminating randomness in server selection.

Benefits of this approach are:

- **Improved Control**: You can explicitly define which AAA server handles validation requests and ensure precise control over server selection.
- **Flexibility**: You can configure either an external AAA server or a local database based on the network's requirements.
- **Enhanced Security**: By specifying the AAA server, you can ensure that only trusted servers are used for validation, reducing the risk of unauthorized access.

Follow these steps to configure an explicit authorization server list using commands:

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Set up the security framework by enabling AAA features for authentication, authorization, and accounting.

Example:

```
Device# aaa new-model
```

Step 4 Configure a RADIUS server to enable centralized authentication, authorization, and accounting for secure and scalable network access control.

a) Specify the RADIUS server name.
b) Specify the RADIUS server parameters.
c) Specify the authorization and encryption key used between the device and the RADIUS daemon running on the RADIUS server by key string encryption.

Example:

```
Device(config)# radius server server-name
Device(config)# address ipv4 address auth-port auth_port_number acct-port acct_port_number
Device(config)# pac key key
Device(config)# end
```

Step 5 Configure the RADIUS group.

a) Create a radius server-group identification.

Note
server-group refers to the server group name. The valid range is from one to 32 alphanumeric characters.

Example:

```
Device(config)# aaa group server radius server-name
Device(config-sg-radius)# server name server-name
Device(config-sg-radius)# exit
```

Step 6 Define a network-level authorization policy for access control using the previously specified server group.

Example:

```
Device(config)# aaa authorization network rogued group group-name
```

Step 7 Create an attribute list to apply custom authentication or authorization attributes.

a) Create an attribute list to specify policies for rogue AP detection and control and enter the attribute list configuration mode.
b) Identify the device with a specific rogue state for network policies.
c) Identify the device with a specific rogue class for network policies.

Example:

```
Device(config)# aaa attribute list aaa-attribute-list-name
Device(config-attr-list)# attribute type rogue-ap-state {alert | contain | threat }
Device(config-attr-list)# attribute type rogue-ap-class {unclassified | malicious | friendly
 }
Device(config-attr-list)# exit
```

Step 8 Define a client with a specific MAC address as a *username* and associate it with the previously defined AAA attribute list. This allows for controlled access or applies policy decisions for that specific client.

Important
For rogue validation, the *username* must be a MAC address in the semicolon format.

Example:

Define a client with a specific MAC address (00:00:00:00:00:00) as a *username* and associate it with an AAA attribute list named **rogue-attributes**. This allows for the use of specific AAA (authentication, authorization,

and accounting) attributes to be applied to this MAC address, typically to control access or apply policy decisions for that client.

```
Device(config)# username 00:00:00:00:00:00 mac aaa attribute list aaa-attribute-list-name
```

Step 9 Display the AAA running configuration for review.

Example:

```
Device(config)# show running-config | include aaa
```

Setting up these configurations ensure that only authorized clients can access the network, and any suspicious activity is flagged and handled appropriately.

Configure authentication and authorization list for 802.1X security (GUI)

Set up 802.1X security by specifying authentication and authorization lists through the GUI.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name** , the **SSID** , and the **WLAN ID**.

Step 4 In the **Security** > **AAA** tab, select an authentication list from the **Authentication List** drop-down list.

Step 5 Click **Apply to Device**.

Configure authentication and authorization list for 802.1X security (CLI)

Configure 802.1X authentication and authorization on a WLAN using commands.

Use this procedure when you need to enforce 802.1X security for a specific wireless network on your device.

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Enter the WLAN configuration sub-mode.

Example:

```
Device(config)# wlan wlan-name wlan-id ssid name
```

- *wlan-name* : is the name of the configured WLAN.
- *wlan-id* : is the wireless LAN identifier. The valid range is from 1 to 512.
- *SSID-name* : is the SSID name which can contain 32 alphanumeric characters.

Note
If you have already configured this command, enter **wlan** *wlan-name* command.

Step 4 Enable authentication list for 802.1x security.

Example:

```
Device(config-wlan)# security dot1x authentication-list authenticate-list-name
```

Step 5 Specify authorization list for 802.1x security.

Example:

```
Device(config-wlan)# security dot1x authorization-list authorize-list-name
```

For more information on the Cisco Catalyst Center, see the Cisco Catalyst Center documentation.

Step 6 Return to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example:

```
Device(config-wlan)# end
```

Configure web authentication for WLAN with split authentication and authorization servers

Configure authentication and authorization list for web authentication (GUI)

Configure authentication and authorization lists to secure web authentication on WLANs using the GUI.

This task ensures only authorized users can access the WLAN, and that access policies are applied during web authentication by assigning the correct authentication and authorization lists.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.

Step 4 In the **Security** > **Layer2** tab, uncheck the **WPAPolicy**, **AES** and **802.1x** check boxes.

Step 5 Check the **MAC Filtering** check box to enable the feature. With MAC Filtering enabled, select the authorization list from the **Authorization List** drop-down list.

Step 6 In the **Security** > **AAA** tab, select the authentication list from the **Authentication List** drop-down list.

Step 7 Click **Apply to Device**.

Configure authentication and authorization list for web authentication (CLI)

Enable web-based authentication and authorization for users connecting to a specified WLAN using commands.

Web authentication allows users to be authenticated through a web browser before gaining access to network resources.

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Enter the WLAN configuration sub-mode.

Example:

```
Device(config)# wlan wlan-name wlan-id ssid-name
```

- *wlan-name* : is the name of the configured WLAN.
- *wlan-id* : is the wireless LAN identifier.
- *SSID-name* : is the SSID name which can contain 32 alphanumeric characters.

Note
If you have already configured this command, enter **wlan** *wlan-name* command.

Step 4 Disable WPA security.

Example:

```
Device(config-wlan)# no security wpa
```

Step 5 Disable security AKM for 802.1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Step 6 Disable WPA2 security.

Example:

```
Device(config-wlan)# no security wpa wpa2
```

Step 7 Enable authentication or authorization list for 802.1x security.

Example:

```
Device(config-wlan)# security web-auth { authentication-list authenticate-list-name |
authorization-list authorize-list-name }
```

Note
When the WPA security, AKM for 802.1x, and WPA2 security settings are not disabled, the system displays the error:

% switch-1:dbm:wireless:web-auth cannot be enabled. Invalid WPA/WPA2 settings.

Step 8 Return to the privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example:

```
Device(config-wlan)# end
```

Verify split authentication and authorization configuration

To view the WLAN details, use this command:

```
Device# show run wlan

wlan wlan-foo 2 foo-ssid
security dot1x authentication-list authc-server-group
security dot1x authorization-list authz-server-group
wlan wlan-bar 3 bar-ssid
security web-auth authentication-list authc-server-group
security web-auth authorization-list authz-server-group
```

To view the AAA authentication and server details, use this command:

```
Device# show run aaa

!
aaa authentication dot1x default group radius
username cisco privilege 15 password 0 cisco
!
!
radius server free-radius-authc-server
 address ipv4 9.2.62.56 auth-port 1812 acct-port 1813
 key cisco
!
radius server cisco-catalyst-center-authz-server
 address ipv4 9.4.62.32 auth-port 1812 acct-port 1813
 pac key cisco
!
!
aaa new-model
aaa session-id common
!
```

To view the authentication and authorization list for 802.1X security, use this command:

```
Device# show wlan name wlan-foo | sec 802.1x

802.1x authentication list name               : authc-server-group
```

```
802.1x authorization list name                : authz-server-group
           802.1x                             : Enabled
```

To view the authentication and authorization list for web authentication, use this command:

```
Device# show wlan name wlan-bar | sec Webauth

Webauth On-mac-filter Failure              : Disabled
Webauth Authentication List Name           : authc-server-group
Webauth Authorization List Name            : authz-server-group
Webauth Parameter Map                      : Disabled
```

Configuration examples

Configure Cisco Embedded Wireless Controller on Catalyst Access Points for authentication with a third-party RADIUS server: example

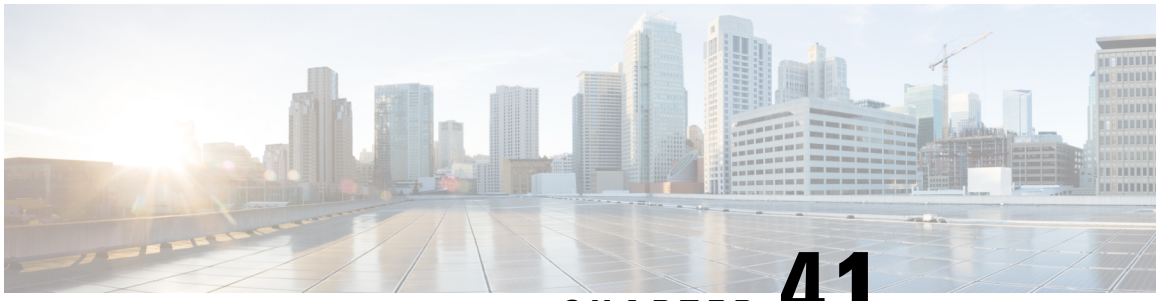
This example shows how to configure Cisco Embedded Wireless Controller on Catalyst Access Points for authentication with a third-party RADIUS server:

```
Device(config)# radius server free-radius-authc-server
Device(config-radius-server)# address ipv4 9.2.62.56 auth-port 1812 acct-port 1813
Device(config-radius-server)# key cisco
Device(config-radius-server)# exit
Device(config)# aaa group server radius authc-server-group
Device(config)# server name free-radius-authc-server
Device(config)# end
```

Configure Cisco Embedded Wireless Controller on Catalyst Access Points for authorization with Cisco ISE or Cisco Catalyst Center: example

This example shows how to configure Cisco Embedded Wireless Controller on Catalyst Access Points for authorization with Cisco ISE or Cisco Catalyst Center:

```
Device(config)# radius server cisco-catalyst-center-authz-server
Device (config-radius-server)# address ipv4 9.4.62.32 auth-port 1812 acct-port 1813
Device (config-radius-server)# pac key cisco
Device (config-radius-server)# exit
Device(config)# aaa group server radius authz-server-group
Device(config)# server name cisco-catalyst-center-authz-server
Device(config)# end
```



CHAPTER 41

Secure LDAP

- Lightweight Directory Access Protocol (LDAP), on page 393

Lightweight Directory Access Protocol (LDAP)

A Lightweight Directory Access Protocol (LDAP) is a client-server protocol that

- allows clients to access and manage directory information stored on servers
- supports user authentication and authorization using operations such as bind and search, and
- operates over TCP/IP on port 389, optionally using Transport Layer Security (TLS) for secure communications.

Transport Layer Security (TLS)

Transport Layer Security (TLS) is an application-level protocol that enables secure data transactions by providing privacy, authentication, and data integrity. TLS relies on certificates, public keys, and private keys to prove the identity of clients.

Certificate Authorities (CAs) issue the certificates.

Each certificate includes these:

- The name of the authority that issued it.
- The name of the entity to which the certificate was issued.
- The public key of the entity.
- The timestamps indicating the expiration date of the certificate.

TLS support for LDAP is described in RFC 2830, which is an extension to the LDAP protocol.

LDAP Operations

Bind

The bind operation authenticates a user to the server and starts a connection with the LDAP server. LDAP is a connection-oriented protocol. The client specifies the protocol version and authentication information.

LDAP supports these binds:

- Authenticated bind: An authenticated bind is performed when a root Distinguished Name (DN) and password are available.
- Anonymous bind: An anonymous bind is performed when no root DN and password are provided.

In LDAP deployments, the search operation is commonly performed before the bind operation. If a password attribute is returned as part of the search operation, the LDAP client can verify the password locally, eliminating the need for an extra bind operation. If the password attribute is not returned, the bind operation can occur later. Another advantage of performing the search operation first is that the LDAP client can use the DN received in the search result as the user DN, instead of constructing it by prefixing the username (cn attribute) with the base DN. All entries stored in an LDAP server have a unique DN.

The DN consists of two parts:

- Relative Distinguished Name (RDN)
- Location in the LDAP server where the record resides.

Most of the entries that you store in an LDAP server will have a name, and the name is frequently stored in the Common Name (cn) attribute. Because every object has a name, most objects stored in an LDAP server use their cn value as the basis for their RDN.

Search

A search operation is used to search the LDAP server. The client specifies the starting point (base DN) of the search, the search scope (either the object, its children, or the subtree rooted at the object), and a search filter.

For authorization requests, the search operation is performed without a bind operation. The LDAP server must be configured with sufficient privileges for the search operation to succeed. These privileges are established during the bind operation.

An LDAP search operation can return multiple entries for a specific user. When this occurs, the LDAP client returns an error code to AAA. To prevent these errors, configure search filters to match only a single entry.

Compare

The compare operation replaces a bind request for authentication and helps maintain the initial bind parameters for the connection.

LDAP Dynamic Attribute Mapping

The Lightweight Directory Access Protocol (LDAP) is a powerful and flexible protocol for communication with AAA servers. LDAP attribute maps provide a method to cross-reference attributes retrieved from a server with Cisco attributes supported by security appliances.

When a user authenticates to a security appliance, the appliance authenticates the server and uses LDAP to retrieve the user's record. The record consists of LDAP attributes associated with fields displayed on the server's user interface. Each attribute retrieved includes a value entered by the administrator responsible for updating user records.

SLDAP prerequisite

This section lists the prerequisite for setting up SLDAP on a wireless controller and describes when to configure certificates.

- If you are using a Transport Layer Security (TLS) secure connection, you must configure X.509 certificates.

SLDAP restrictions

- LDAP referrals are not supported.
- The LDAP server's unsolicited messages or notifications are not handled.
- LDAP authentication is not supported for interactive terminal sessions.

Configure SLDAP (CLI)

Set up Secure LDAP (SLDAP) for authentication and directory queries on the wireless controller using commands.

Before you begin

Ensure you have administrator access to the controller. Gather LDAP server details (IP address, base DN, credentials).

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Define a Lightweight Directory Access Protocol (LDAP) server and enter the LDAP server configuration mode.

Example:

```
Device(config)# ldap server server-name
```

Step 4 Specify the LDAP server IP address using IPv4.

Example:

```
Device(config-ldap-server)# ipv4 ipv4-address
```

Step 5 Specify the number of seconds the or the embedded wireless controller waits for a reply to a LDAP request before retransmitting the request.

Example:

```
Device(config-ldap-server)# timeout retransmit seconds
```

Step 6 Specify a shared secret text string used between the or the embedded wireless controller and a LDAP server.

Example:

```
Device(config-ldap-server)# bind authenticate root-dn
CN=ldapipv6user,CN=Users,DC=ca,DC=ssh2,DC=com password Cisco12345
```

Use the **0** line option to configure an unencrypted shared secret.

Use the **7** line option to configure an encrypted shared secret.

Step 7 Specify the base Distinguished Name (DN) of the search.

Example:

```
Device(config-ldap-server)# base-dn string CN=Users,DC=ca,DC=ssh2,DC=com
```

Step 8 Configure LDAP to initiate the TLS connection and specify the secure mode.

Example:

```
Device(config-ldap-server)# mode secure no- negotiation
```

Step 9 Return to the privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example:

```
Device(config-ldap-server)# end
```

Configure a AAA server group (GUI)

Configure a AAA server group using the GUI for RADIUS, TACACS+, or LDAP server hosts using the GUI.

Configuring a device to use AAA server groups helps you group existing server hosts, select a subset of those hosts, and assign them for a particular service. A server group is used with a global server-host list. The server group lists the IP addresses of the selected server hosts.

You can create these server groups:

Before you begin

Procedure

Step 1 RADIUS

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **RADIUS**.
b) Click the **Add** button. The **Create AAA Radius Server Group** dialog box appears.
c) Enter a name for the RADIUS server group in the **Name** field.
d) Select a desired delimiter from the **MAC-Delimiter** drop-down list. The available options are colon, hyphen, and single-hyphen.
e) Select a desired filter from the **MAC-Filtering** drop-down list. The available options are mac and Key.
f) Enter a value in the **Dead-Time (mins)** field to make a server non-operational. You must specify a value between 1 and 1440.
g) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
h) Click the **Save & Apply to Device** button.

Step 2 TACACS+

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **TACACS+**.
b) Click the **Add** button. The **Create AAA Tacacs Server Group** dialog box appears.
c) Enter a name for the TACACS server group in the **Name** field.

d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
e) Click the **Save & Apply to Device** button.

Step 3 LDAP

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **LDAP**.
b) Click the **Add** button. The **Create AAA Ldap Server Group** dialog box appears.
c) Enter a name for the LDAP server group in the **Name** field.
d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
e) Click the **Save & Apply to Device** button.

Configure a AAA server group (CLI)

Establish a AAA server group for centralized authentication, authorization, and accounting using commands.

AAA server groups allow you to centrally manage user access and policy enforcement with protocols such as LDAP, RADIUS, or TACACS+.

Procedure

Step 1 Enable the privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Enable AAA.

Example:

```
Device(config)# aaa new-model
```

Step 4 Define the AAA server group with a group name and enter the LDAP server group configuration mode.

Example:

```
Device(config)# aaa group server ldap group-name
```

All members of a group are of the same type: RADIUS, LDAP, or TACACS+.

Step 5 Associate a particular LDAP server with the defined server group.

Example:

```
Device(config-ldap-sg)# server server-name
```

Each security server is identified by its IP address and UDP port number.

Step 6 Exit the LDAP server group configuration mode.

Example:

```
Device(config-ldap-sg)# exit
```

Configure search and bind operations for authentication requests (CLI)

Enable LDAP-based authentication by defining how the controller searches and binds user credentials during authentication requests using commands.

Configuring search and bind operations determines whether the LDAP server first binds the user and then performs a search, or compares credentials without binding. This affects how authentication requests are processed on the controller.

Procedure

Step 1 Enable the privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Enable AAA.

Example:

```
Device(config)# aaa new-model
```

Step 4 Define a Lightweight Directory Access Protocol (LDAP) server and enter the LDAP server configuration mode.

Example:

```
Device(config)# ldap server server-name
```

Step 5 Configure the sequence of search and bind operations for an authentication request.

Example:

```
Device(config-ldap-server)# authentication bind-first
```

Step 6 Replace the bind request with the compare request for authentication.

Example:

```
Device(config-ldap-server)# authentication compare
```

Step 7 Exit the LDAP server group configuration mode.

Example:

```
Device(config-ldap-server)# exit
```

Configure a dynamic attribute map on a SLDAP server (CLI)

Configure dynamic LDAP attribute maps to ensure seamless interoperability between user-defined attribute names and Cisco attribute requirements on a wireless controller using commands.

You must create LDAP attribute maps that map your existing user-defined attribute names and values to Cisco attribute names and values that are compatible with the security appliance. You can then bind these attribute maps to LDAP servers or remove them as required.



Note To use the attribute mapping features correctly, make sure you understand Cisco LDAP and user-defined attribute names and values.

Before you begin

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure a dynamic LDAP attribute map and enter the attribute-map configuration mode.

Example:

```
Device(config)# ldap attribute-map map-name
```

Step 4 Define an attribute map.

Example:

```
Device(config-attr-map)# map type ldap-attr-type aaa-attr-type
```

Step 5 Exit the attribute-map configuration mode.

Example:

```
Device(config-attr-map)# exit
```

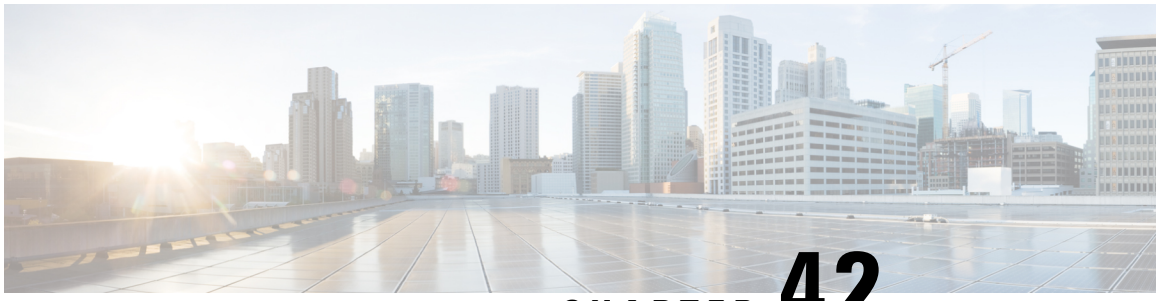
Verify the SLDAP configuration

To view details about the default LDAP attribute mapping, use this command:

```
Device# show ldap attributes
```

To view the LDAP server state information and various other counters for the server, use this command:

```
Device# show ldap server
```



CHAPTER 42

RADIUS DTLS

- RADIUS DTLS, on page 401

RADIUS DTLS

RADIUS DTLS is a client or server protocol that

- provides centralized security for users attempting to gain management access to a network
- delivers a complete Authentication, Authorization, and Accounting (AAA) solution, and
- uses RADIUS-over-DTLS (UDP) for secure communication.

RADIUS DTLS components and configuration

The Remote Authentication Dial-In User Service (RADIUS) is a widely deployed authentication and authorization protocol.

RADIUS DTLS includes these components:

- **RADIUS DTLS Port**: Used for authentication and accounting with default DTLS server port 2083
- **Shared Secret**: Uses RADIUS/DTLS as the shared secret when DTLS is enabled for a specific server
- **PAC Handling**: Downloads PAC from ISE for CTS communication and encrypts CTS attributes with the PAC key
- **Session Management**: Depends on DTLS server responses and closes sessions for ideal timeout or invalid responses
- **Load Balancing**: Supports multiple DTLS servers and load balancing methods

You can change the RADIUS DTLS port number using **DTLS port** *port_number*. For more information, see the Configuring RADIUS DTLS Port Number section.

RADSEC consists of two types: RADIUS-over-TLS (using TCP) and RADIUS-over-DTLS (using UDP). Cisco IOS-XE support RADIUS-over-DTLS (UDP) but does not support RADIUS-over-TLS (TCP), as outlined in RFC 7360.

RADIUS DTLS timeout and retry configuration includes these parameters:

- **Connection Timeout**: Default connection timeout is 5 seconds

- **Connection Retries**: Default connection retries is 5
- **Idle Timeout**: Default idle timeout is 60 seconds

After the encrypted RADIUS packet is sent, you need to start the retransmission timer. If you do not get a response before the retransmission timer expires, the packet is re-encrypted and re-transmitted. You can continue for number of times as per the **DTLS retries** configuration or till the default value. Once the number of tries exceeds the limit, the server becomes unavailable and responses are sent back to the AAA clients.

As the RADIUS DTLS is UDP based, you need to retry the connection after a specific timeout interval for a specific number of retries. After all retries are exhausted, the DTLS connection performs these actions:

- Is marked as unsuccessful.
- Looks up for the next available server for processing the RADIUS requests.

When the idle timer expires and no transactions exists since the last idle timeout, the DTLS session remains closed. After you establish the DTLS session, you can start the idle timer. If you start the idle timer for 30 seconds and one of the RADIUS DTLS packet is sent, then after 30 seconds, the idle timer expires and checks for number of RADIUS DTLS transactions. If the idle timer value exceeds zero, the idle timer resets the transaction counter and restarts the timer.

You can configure RADIUS servers with and without DTLS. It is recommended to create AAA server groups with DTLS enabled servers and non-DTLS servers. However, you will not find any such restriction while configuring AAA server groups.

Suppose you choose a DTLS server, the DTLS server establishes connection and RADIUS request packet is sent to the DTLS server. If the DTLS server does not respond after all RADIUS retries, it would fall over to the next configured server in the same server group. If the next server is a DTLS server, the processing of the RADIUS request packet continues with the next server. If the next server is a non-DTLS server, the processing of RADIUS request packet does not happen in that server group. Then the server group failover occurs and the same sequence continues with the next server group, if the next server group is available.

Note You need to use either only DTLS or non-DTLS servers in a server group.

Requirement: Port RADIUS DTLS support to BINOS platforms

Support for IOS and BINOS AAA

Port RADIUS DTLS support to BINOS after completing implementation in IOS, since the AAA server runs on both IOS and BINOS platforms.

Configure RADIUS DTLS server (CLI)

Enable secure RADIUS communication using DTLS encryption on a wireless controller.

DTLS (Datagram Transport Layer Security) provides secure communication for RADIUS authentication in wireless networks. This configuration is used to protect RADIUS traffic between the wireless controller and RADIUS server.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure DTLS parameters.

Example:

```
Device(config-radius-server)# dtls
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS server is now configured with DTLS security, enabling encrypted communication for authentication requests.

Configure RADIUS DTLS connection timeout

Set the timeout value for DTLS connections to RADIUS servers to ensure proper connection management and security.

DTLS (Datagram Transport Layer Security) connection timeout configuration is necessary when using encrypted communication between the wireless controller and RADIUS servers. This setting determines how long the controller waits before timing out a DTLS connection attempt.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure RADIUS DTLS connection timeout.

Example:

```
Device(config-radius-server)# dtls connectiontimeout timeout-value
```

Example:

```
Device(config-radius-server)# dtls connectiontimeout 1
```

The *timeout-value* refers to the DTLS connection timeout value. The valid range is from 1 to 65535.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS connection timeout is now configured with the specified timeout value for the designated RADIUS server.

Configure RADIUS DTLS idle timeout

Set up RADIUS DTLS idle timeout to control the duration before idle DTLS connections are terminated.

RADIUS DTLS idle timeout configuration is necessary to manage connection resources and ensure proper cleanup of inactive DTLS sessions between the wireless controller and RADIUS server.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure RADIUS DTLS idle timeout.

Example:

```
Device(config-radius-server)# dtls idletimeout idle-timeout-value
```

Example:

```
Device(config-radius-server)# dtls idletimeout 2
```

The DTLS idle timeout value valid range is from 1 to 65535.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS idle timeout is configured and will terminate idle DTLS connections after the specified timeout period.

Configure source interface for RADIUS DTLS server

Configure the source interface for RADIUS DTLS server to establish secure communication between the wireless controller and RADIUS server.

When configuring RADIUS authentication with DTLS encryption, you need to specify the source interface that the wireless controller will use to communicate with the RADIUS server. This ensures proper routing and security for authentication traffic.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure source interface for RADIUS DTLS server.

Example:

```
Device(config-radius-server)# dtls ip radius source-interface Ethernet-Internal
interface-number
```

Example:

```
Device(config-radius-server)# dtls ip radius source-interface Ethernet-Internal 0
```

The *interface-number* refers to the Ethernet-Internal interface number. The default value is 0.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The source interface for RADIUS DTLS server is now configured, enabling secure RADIUS communication through the specified Ethernet-Internal interface.

Configure RADIUS DTLS port number (CLI)

Configure the RADIUS DTLS port number to establish secure RADIUS communication using DTLS protocol.

RADIUS DTLS provides secure authentication and accounting communication between the wireless controller and RADIUS server. Configure the DTLS port when you need to specify a custom port number for DTLS communication.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure RADIUS DTLS port number.

Example:

```
Device(config-radius-server)# dtls port port-number
```

Example:

```
Device(config-radius-server)# dtls port 2
```

The *port-number* refers to the DTLS port number.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS port number is configured and the controller can now use the specified port for secure DTLS communication with the RADIUS server.

Configure RADIUS DTLS connection retries (CLI)

Set the number of RADIUS DTLS connection retry attempts to ensure reliable authentication server connectivity.

RADIUS DTLS connection retries help maintain authentication service availability when network conditions cause connection failures. Configure this setting when you need to control how many times the wireless controller attempts to reconnect to the RADIUS server before giving up.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure RADIUS connection retries.

Example:

```
Device(config-radius-server)# dtls retries retry-number
```

Example:

```
Device(config-radius-server)# dtls retries 3
```

The *retry-number* refers to the DTLS connection retries. The valid range is from 1 to 65535.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS connection retry count is configured and the wireless controller will attempt the specified number of reconnections before failing over or giving up.

Configure RADIUS DTLS trustpoint (CLI)

Establish secure DTLS communication between the wireless controller and RADIUS server by configuring appropriate trustpoints for authentication.

RADIUS DTLS trustpoints provide secure communication channels for RADIUS authentication traffic. Configure both client and server trustpoints to ensure proper certificate validation during DTLS handshakes.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Specify the RADIUS server name.

Example:

```
Device(config)# radius server server-name
```

Example:

```
Device(config)# radius server R1
```

Step 4 Configure trustpoint for client and server.

Example:

```
Device(config-radius-server)# dtls trustpoint client client-trustpoint-name dtls
```

Example:

```
Device(config-radius-server)# dtls trustpoint server server-trustpoint-name dtls
```

Example:

```
Device(config-radius-server)# dtls trustpoint client client1 dtls
```

Example:

```
Device(config-radius-server)# dtls trustpoint server server1 dtls
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS trustpoints are now configured, enabling secure DTLS communication between the wireless controller and RADIUS server for authentication traffic.

Configure RADIUS DTLS match-server-identity

This task configures RADIUS DTLS match-server-identity parameters to establish secure authentication between the wireless controller and RADIUS server.

RADIUS DTLS (Datagram Transport Layer Security) provides encrypted communication for RADIUS authentication. The match-server-identity feature validates the RADIUS server's certificate against its hostname or IP address to ensure secure connections.

Procedure

Step 1 Configure the RADSEC certification validation parameters.

Example:

```
dtls match-server-identity hostname <name>
```

Step 2 Configure the RADSEC certification validation parameters.

Example:

```
dtls match-server-identity ip-address <IPv4 or IPv6>
```

The RADIUS DTLS match-server-identity parameters are configured, enabling secure certificate validation for RADSEC communication.

Configure DTLS Dynamic Author (CLI)

Enable DTLS support for RADIUS dynamic authorization (RFC 3576) on the wireless controller.

DTLS Dynamic Author provides secure communication for RADIUS Change of Authorization (CoA) and disconnect messages. This configuration is needed when implementing dynamic policy changes and session management in wireless networks.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure local server profile for RFC 3576 support.

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 4 Configure DTLS source parameters.

Example:

```
Device(config-locsvr-da-radius)# dtls
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

DTLS Dynamic Author is now configured on the wireless controller, enabling secure RADIUS dynamic authorization communications.

Enable DTLS for client (CLI)

Enable DTLS encryption for client communications to secure RADIUS dynamic authorization messages.

DTLS (Datagram Transport Layer Security) provides encryption for UDP-based communications between the wireless controller and RADIUS clients. This configuration enhances security for RFC 3576 dynamic authorization operations.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure local server profile for RFC 3576 support.

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 4 Enable DTLS for the client.

Example:

```
Device(config-locsvr-da-radius)# client IP-address dtls
```

Example:

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

DTLS is now enabled for the specified client, providing encrypted communication for RADIUS dynamic authorization messages.

Configure client trustpoint for DTLS (CLI)

Establish secure DTLS communication between the wireless controller and RADIUS dynamic authorization clients.

Client trustpoint configuration for DTLS is required when implementing secure communication for RFC 3576 dynamic authorization services on wireless controllers.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure local server profile for RFC 3576 support.

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 4 Configure client trustpoint for DTLS.

Example:

```
Device(config-locsvr-da-radius)# client IP-address dtls client-tp client-tp-name
```

Example:

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls client-tp client_tp_name
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The client trustpoint for DTLS is now configured, enabling secure communication between the wireless controller and the specified RADIUS dynamic authorization client.

Configure DTLS idle timeout (CLI)

Set the DTLS idle timeout interval to control when inactive DTLS connections are terminated.

DTLS idle timeout configuration is necessary to manage inactive DTLS connections and optimize resource utilization on wireless controllers.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure local server profile for RFC 3576 support.

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 4 Configure DTLS idle time.

Example:

```
Device(config-locsvr-da-radius)# client IP_addr dtls idletimeout timeout-interval
{client-tp client_tp_name | server-tp server_tp_name}client IP-address dtls idletimeout
 timeout-interval client-tp client-tp-name
```

Example:

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 62 client-tp dtls_ise
```

The *timeout-interval* refers to the idle timeout interval. The valid range is from 60 to 600.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The DTLS idle timeout is now configured and will terminate inactive DTLS connections after the specified timeout interval.

Configure server trustpoint for DTLS (CLI)

Enable secure DTLS communication between the wireless controller and RADIUS clients by configuring a server trustpoint.

Server trustpoints are required when implementing DTLS (Datagram Transport Layer Security) for RFC 3576 Change of Authorization (CoA) messages between RADIUS clients and the wireless controller. This ensures encrypted and authenticated communication.

Procedure

Step 1 Enter privileged EXEC mode.

Example:

```
Device# enable
```

Step 2 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 3 Configure local server profile for RFC 3576 support.

Example:

```
Device(config)# aaa server radius dynamic-author
```

Step 4 Configure server trust point for the DTLS client.

Example:

```
Device(config-locsvr-da-radius)# client IP-address dtls server-tp server-trustpoint-name
```

Example:

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls server-tp dtls_client
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The server trustpoint for DTLS is now configured, enabling secure communication between the wireless controller and the specified RADIUS client for Change of Authorization messages.

Verify RADIUS DTLS server configuration

To view information about the DTLS enabled servers, use this command:

```
Device# show aaa servers
DTLS: Packet count since last idletimeout 1,
Send handshake count 3,
Handshake Success 1,
Total Packets Transmitted 1,
Total Packets Received 1,
Total Connection Resets 2,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 2,
Connection Reset due to Malformed packet 0,
```

Clear RADIUS DTLS specific statistics

Clear RADIUS DTLS specific statistics to reset monitoring counters and start fresh data collection.

Use this task when you need to reset DTLS server statistics for monitoring purposes or troubleshooting.

Procedure

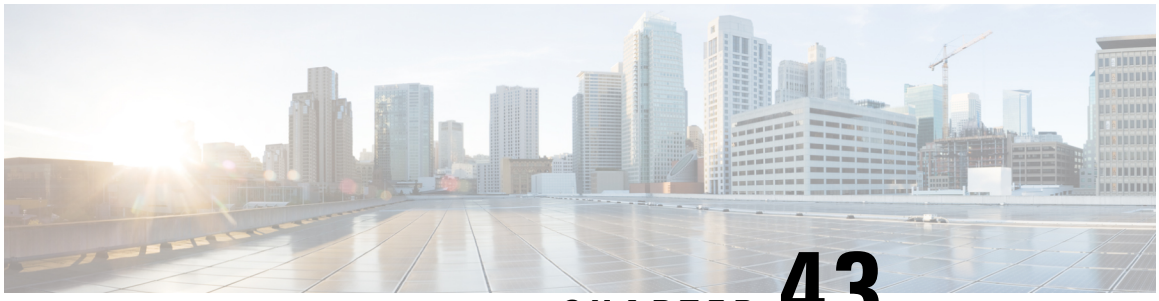
Clear the RADIUS DTLS specific statistics.

Example:

```
Device# clear aaa counters servers radius {<server-id> | all}
```

Note
Here, *server-ID* refers to the server ID displayed by **show aaa servers**. The valid range is from 0 to 2147483647.

The RADIUS DTLS specific statistics are cleared and reset to zero.



CHAPTER 43

MAC Filtering

- MAC Filtering, on page 415

MAC Filtering

You can configure the embedded wireless controller to authorize clients based on the client MAC address by using the MAC filtering feature.

When MAC filtering is enabled, the embedded wireless controller uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. The embedded wireless controller sends the authentication server a RADIUS-access/request frame with a username and password based on the client MAC address as soon as it gets the association request from the client. If authorization succeeds, the embedded wireless controller sends a successful association response to the client. If authorization fails, the embedded wireless controller rejects the client association.

Clients that were authorized with MAC filtering can be re-authenticated through the WLAN session timeout feature.

MAC Filtering Configuration Guidelines

- MAC filtering authentication occurs at the 802.11 association phase and delays the association response until authentication is done. If you use a RADIUS server for MAC filtering, it is advised to keep a low latency between the controller and the RADIUS server. When latency is too high, the client might timeout while waiting for the association response.
- MAC filtering can be combined with other authentication methods such as 802.1X, Pre-Shared Key or it can be used alone.
- MAC addresses can be spoofed and MAC filtering does not consist in a security measure.
- Many clients can use a private MAC address to connect and change it at every session, therefore making it harder to identify devices through their MAC address.

Note If wlan-profile-name is configured for a user, guest user authentication is allowed only from that WLAN.

If wlan-profile-name is not configured for a user, guest user authentication is allowed on any WLAN.

If you want the client to connect to SSID1, but not to SSID2 using mac-filtering, ensure that you configure **aaa-override** in the policy profile.

In the following example, when a client with MAC address 112233440001 tries to connect to a WLAN, the request is sent to the local RADIUS server, which checks the presence of the client MAC address in its attribute list (FILTER_1 and FILTER_2). If the client MAC address is listed in an attribute list (FILTER_1), the client is allowed to join the WLAN (WLAN_1) that is returned as *ssid attribute* from the RADIUS server. The client is rejected, if the client MAC address is not listed in the attribute list.

Local RADIUS Server Configuration

```
!Configures an attribute list as FILTER_2
aaa attribute list FILTER_2
!Defines an attribute type that is to be added to an attribute list.
attribute type ssid "WLAN_2"

!Username with the MAC address is added to the filter
username 112233440001 mac aaa attribute list FILTER_2

!
aaa attribute list FILTER_1
attribute type ssid "WLAN_1"
username 112233440001 aaa attribute list FILTER_1
```

Controller Configuration

```
! Sets authorization to the local radius server
aaa authorization network MLIST_MACFILTER local

!A WLAN with the SSID WLAN_2 is created and MAC filtering is set along with security
parameters.
wlan WLAN_2 2 WLAN_2
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers

!WLAN with the SSID WLAN_1 is created and MAC filtering is set along with security parameters.
wlan WLAN_1 1 WLAN_1
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers aes
no security wpa akm dot1x
security web-auth
security web-auth authentication-list WEBAUTH

! Policy profile to be associated with the above WLANs
wireless profile policy MAC_FILTER_POLICY
aaa-override
vlan 504
no shutdown
```

Configuring MAC Filtering for Local Authentication (CLI)

Follow the procedure given below to configure MAB for local authentication.

Before you begin

Configure AAA local authentication.

Configure the username for WLAN configuration (local authentication) using **username** *mac-address* **mac** command.

Note The mac-address must be in the following format: *abcdabcdabcd*

Procedure

Step 1 **wlan** *profile-name wlan-id*

Example:

```
wlan CR1_SSID_mab-local-default 1 CR1_SSID_mab-local-default
```

Specifies the WLAN name and ID.

Step 2 **mac-filtering default**

Example:

```
Device(config-wlan)# mac-filtering default
```

Sets MAC filtering support for the WLAN.

Step 3 **no security wpa**

Example:

```
Device(config-wlan)# no security wpa
```

Disables WPA secuirty.

Step 4 **no security wpa akm dot1x**

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Disables security AKM for dot1x.

Step 5 **no security wpa wpa2**

Example:

```
Device(config-wlan)# no security wpa wpa2
```

Disables WPA2 security.

Step 6 **no security wpa wpa2 ciphers aes**

Example:

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

Disables WPA2 ciphers for AES.

Step 7 **no shutdown**

Example:

```
Device(config-wlan)# no shutdown
```

Enables the WLAN.

Configuring MAC Filtering (GUI)

Before you begin

Configure AAA external authentication.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **WLANs**.
Step 2 On the **Wireless Networks** page, click the name of the WLAN.
Step 3 In the **Edit WLAN** window, click the **Security** tab.
Step 4 In the **Layer2** tab, check the **MAC Filtering** check box to enable the feature.
Step 5 With MAC Filtering enabled, choose the **Authorization List** from the drop-down list.
Step 6 Save the configuration.

Configuring MAB for External Authentication (CLI)

Follow the procedure given below to configure MAB for external authentication.

Before you begin

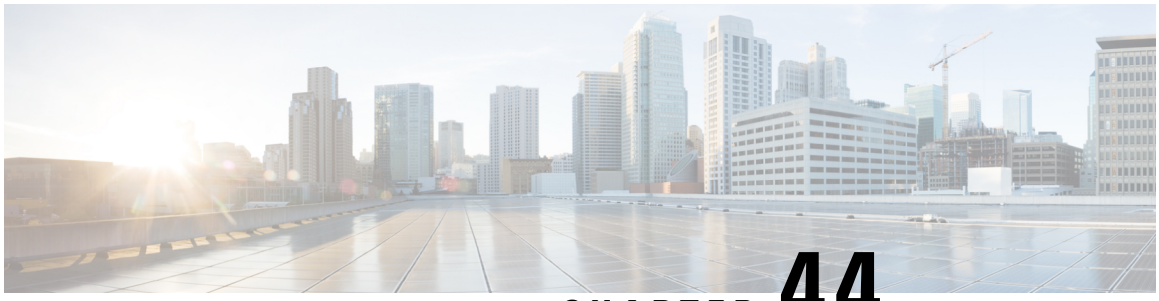
Configure AAA external authentication.

Procedure

	Command or Action	Purpose
Step 1	**wlan** *wlan-name wlan-id ssid-name* **Example:** `wlan CR1_SSID_mab-ext-radius 3 CR1_SSID_mab-ext-radius`	Specifies the WLAN name and ID.
Step 2	**mac-filtering** *list-name* **Example:** `Device(config-wlan)# mac-filtering ewlc-radius`	Sets the MAC filtering parameters. Here, *ewlc-radius* is an example for the *list-name*
Step 3	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA secuirty.
Step 4	**no security wpa akm dot1x** **Example:**	Disables security AKM for dot1x.

	Command or Action	Purpose
	`Device(config-wlan)# no security wpa akm dot1x`	
Step 5	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.
Step 6	**mab request format attribute** {**1 groupsize** *size* **separator** *separator* [**lowercase** | **uppercase**] | **2** {**0** | **7** | **LINE**} **LINE** *password* | **32 vlan access-vlan**} **Example:** `Device(config)# mab request format attribute 1 groupsize 4 separator`	Optional. Configures the delimiter while using MAC filtering in a WLAN. Here, **1**- Specifies the username format used for MAB requests. **groupsize** *size*- Specifies the number of hex digits per group. The valid values range from 1 to 12. **separator** *separator*- Specifies how to separate groups. The separators are comma, semicolon, and full stop. **lowercase**- Specifies the username in lowercase format. **uppercase**- Specifies the username in uppercase format. **2**- Specifies the global password used for all the MAB requests. **0**- Specifies the unencrypted password. **7**- Specifies the hidden password. **LINE**- Specifies the encrypted or unencrypted password. *password*- LINE password. **32**- Specifies the NAS-Identifier attribute. **vlan**- Specifies a VLAN. **access-vlan**- Specifies the configured access VLAN.
Step 7	**no security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.
Step 8	**no shutdown** **Example:**	Enables the WLAN.

	Command or Action	Purpose
	`Device(config-wlan)# no shutdown`	



CHAPTER 44

Dynamic Frequency Selection

• Dynamic frequency selection, on page 421

Dynamic frequency selection

Dynamic Frequency Selection (DFS) is a process that

- detects radar signals on DFS-enabled 5.0-GHz (802.11a/h) radios
- automatically sets the frequency to avoid interference with radar signals, and
- ensures radios configured for use in a regulatory domain do not interfere with radar systems.

Flex DFS behavior

In normal DFS, when a radar signal is detected on any of the channels in the 40-MHz or 80-MHz bandwidth, the whole channel is blocked. With Flex DFS, if the radar signals are not detected on the secondary channel, the AP is moved to a secondary channel with a reduction in the bandwidth, usually, by half.

Configure dynamic frequency selection (GUI)

Before you begin

Follow these steps to configure Dynamic Frequency Selection using GUI:

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**

Step 2 Choose a profile.

Step 3 In **General** tab, check the **Full sector DFS status** check box.

Step 4 Click **Update & Apply to Device**.

Configure dynamic frequency selection

Configure DFS to enable automatic radar detection and channel switching on 5 GHz wireless networks to comply with regulatory requirements.

Dynamic Frequency Selection (DFS) is required for 5 GHz wireless operations to detect radar signals and automatically switch channels when interference is detected. This ensures compliance with regulatory standards while maintaining network connectivity.

Before you begin

- The corresponding AP must be on one of the DFS channels.
- Shut down the radio before applying the configuration changes.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Disable the 802.11a Dynamic Transmit Power Control (DTPC) setting.

Example:

```
Device(config)# no ap dot11 5ghz dtpc
```

Step 3 Configure the 802.11h channel switch mode.

Example:

```
Device(config)# ap dot11 5ghz channelswitch mode mode-num
```

Example:

```
Device(config)# ap dot11 5ghz channelswitch mode 1
```

Step 4 Configure the 802.11h power-constraint value.

Example:

```
Device(config)# ap dot11 5ghz power-constraint value
```

Example:

```
Device(config)# ap dot11 5ghz power-constraint 12
```

Step 5 Configure nonoccupancy time for the radar interference channel.

Example:

```
Device(config)# ap dot11 5ghz smart-dfs
```

Dynamic Frequency Selection is now configured, enabling automatic radar detection and channel switching to maintain regulatory compliance on 5 GHz wireless networks.

DFS verification

Use the following commands to verify the DFS configuration:

To display the 802.11h configuration, use the following command:

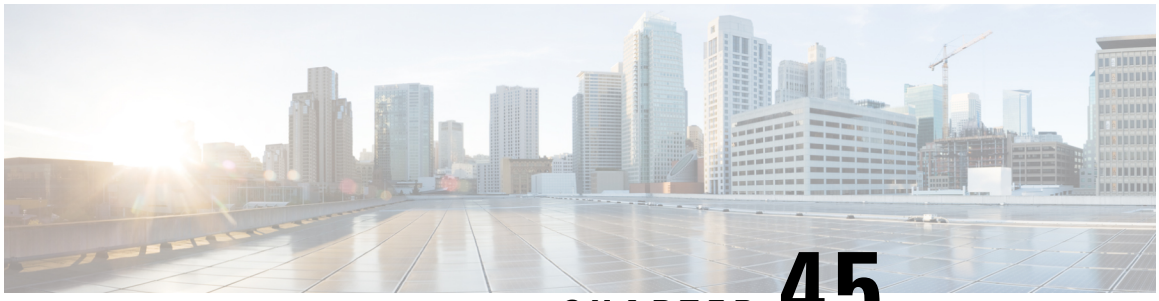
```
Device# show wireless dot11h
```

To display the auto-rF information for 802.11h configuration, use the following command:

```
Device# show ap auto-rf dot11 5ghz
```

To display the auto-rF information for a Cisco AP, use the following command:

```
Device# show ap name ap1 auto-rf dot11 5gh
```



CHAPTER 45

Managing Rogue Devices

- Rogue devices and detection, on page 425
- Rogue Location Discovery Protocol (RLDP), on page 440
- Wireless Service Assurance (WSA) rogue events, on page 447
- Rogue PMF containment, on page 449

Rogue devices and detection

A rogue device is any unauthorized network device.

A rogue device can:

- disrupt wireless LAN operations by hijacking legitimate clients,
- facilitate attacks such as plaintext, denial-of-service, and man-in-the-middle attacks on wireless networks, and
- pose serious security risks by allowing unauthorized access, interception, and breaches inside the corporate firewall.

Risks and impact on network security

Rogue access points are a common form of rogue devices. Hackers can use rogue access points to capture sensitive information such as usernames and passwords. By transmitting a series of Clear to Send (CTS) frames, a rogue access point can mimic a legitimate access point. This action instructs a specific client to transmit while forcing other clients to wait, which prevents legitimate clients from accessing network resources.

Ban rogue access points from the air space to protect users and maintain network integrity.

Because rogue access points are inexpensive and available, staff may connect unauthorized access points to existing LANs. This can create ad hoc wireless networks without approval from IT departments.

These rogue access points can create serious security risks because they may connect inside the corporate firewall. If security settings are disabled on these devices, unauthorized users can intercept network traffic and hijack client sessions. When wireless users connect to rogue access points within the enterprise network, the risk of a security breach increases.

Rogue client status change

The controller marks the rogue client as a threat if a wireless client in the RUN state has the same MAC address.

Restrictions on rogue detection

- Rogue containment is not supported on DFS channels.

Rogue AP containment and attack signatures

A rogue AP containment and attack signature is a set of wireless network security mechanisms that

- detect and contain unauthorized rogue wireless APs,
- use automated or manual methods to select the best access point for containment actions, and
- define specific attack signatures, including behaviors of rogue AP impersonators and new threats based on beacon frames.

Containment operation

A rogue access point is moved to a contained state either automatically or manually. The controller selects the best available access point (AP) for containment and pushes the containment information to it. Each AP stores its list of containments per radio.

For auto-containment, you can configure the controller to use only APs in monitor mode. Containment operations occur through two primary methods:

- The container AP periodically reviews its containment list and sends unicast containment frames. For rogue AP containment, these are sent only if a rogue client is associated.
- When the system detects contained rogue activity, the AP immediately transmits containment frames.

Individual rogue containment involves sending a sequence of unicast disassociation and deauthentication frames.

Attack signature examples

These signatures help identify sophisticated attack methods targeting wireless networks and support more effective containment and remediation strategies.

-
-

Cisco Prime Infrastructure interaction and rogue detection

A rogue access point classification rule is a network security mechanism that

- applies predefined criteria to evaluate and categorize access points detected on the network,
- determines the current state of each detected access point (such as Friendly, Malicious, Internal, or External), and

- defines when and how the controller communicates rogue events to Cisco Prime Infrastructure.

How Cisco Prime Infrastructure interacts with the controller for rogue detection

Cisco Prime Infrastructure interacts with controllers to detect and manage rogue access points according to the classification rules set on the controllers. The rule-based classification enables Cisco Prime Infrastructure to receive detailed trap notifications when rogue access point events occur. Key interactions and behaviors include:

- The controller uses its configured classification rules to determine the state of each rogue access point.
- When certain rogue access point events occur (such as state changes or rogue entry removal), the controller sends traps (notifications) to Cisco Prime Infrastructure.
- Trap notifications depend on both the rogue state and the event type, including these cases:
  - If an unknown access point moves to the Friendly state for the first time and the rogue state is Alert, a trap is sent.
  - No trap is sent if the rogue state is Internal or External.
  - If a rogue entry is removed after timeout, the controller sends a trap for entries classified as Malicious (Alert, Threat) or Unclassified (Alert).
  - The controller does not remove rogue entries with the following states: Contained, Contained Pending, Internal, and External.
- When a new, unknown access point is detected and moves to the Friendly state under Alert, Cisco Prime Infrastructure receives a notification.
- If a rogue entry with a Malicious or Unclassified (Alert) state is removed after a timeout, a trap is generated and sent to Cisco Prime Infrastructure.
- Rogue entries in Contained, Contained Pending, Internal, and External states are retained by the controller and not removed automatically, so no removal trap is sent for them.

AP impersonation detection

An AP impersonation is a wireless security attack that

- allows a rogue device to masquerade as a legitimate AP
- enables attackers to intercept and manipulate wireless communication between client devices and the network, and
- threatens the confidentiality, integrity, and security of wireless network traffic.

Detection methods

The various methods to detect AP impersonation are:

- You can detect AP impersonation if a managed AP reports itself as Rogue. This method is always enabled and does not require configuration.

- AP impersonation detection uses Management Frame Protection (MFP).

Management Frame Protection (MFP)-based detection

Infrastructure MFP protects 802.11 session management by adding Message Integrity Check (MIC) elements to management frames sent by APs (not clients). Other APs in the network then validate these management frames.

- If infrastructure MFP is enabled, managed APs check whether the MIC elements are present and valid.
- If either condition is not met, the managed AP sends rogue AP reports with an updated AP authentication failure counter field.

AP authentication-based detection

When you enable AP authentication, the controller creates an AP domain secret and shares it with all APs in the same network. This process enables APs to authenticate each other.

- An AP authentication information element is attached to beacon and probe response frames.
- If the AP authentication information element has an incorrect signature, an off timestamp, or the information element is missing, the AP that detects the condition increments the **AP authentication failure count** field.
- An impersonation alarm is raised after the **AP authentication failure count** field exceeds its threshold.
- The rogue AP is classified as **Malicious** with the state **Threat**.

Run the **show wireless wps rogue ap detail** command to see when AP impersonation is detected as a result of authentication errors.

Configuration notes

- Run the **CCX Aironet-IESupport** command in all WLAN procedures to prevent the BSSID from being detected as a rogue.
- For AP impersonation detection, Network Time Protocol (NTP) must be enabled under the AP profile. CAPWAP-based time is not sufficient.

Rogue detection security level

A rogue detection security level is a configuration preset that

- determines the sensitivity and scope of rogue wireless device detection
- restricts or allows configuration of specific detection parameters, and
- provides predefined or customizable options for different deployment needs.

- Rogue detection: identifies unauthorized or unknown wireless devices in a network environment.
- Security level: specifies a preset combination of parameters for rogue detection.

The system provides four rogue detection security levels.

- Critical: Provides basic rogue detection for highly sensitive deployments. Fixed configuration parameters ensure maximum security and consistency.
- High: Provides basic rogue detection suitable for medium-scale environments. Several parameters are fixed to balance protection and operational simplicity.
- Low: Provides basic rogue detection suitable for small-scale deployments. Fixed parameters provide easy management.
- Custom: The default security level. You can fully configure all rogue detection parameters to suit any environment.



Note To modify all parameters, select the Custom security level. The critical, high, and low levels have fixed settings.

Table 17: Rogue Detection: Predefined Levels

Parameter	Critical	High	Low
Cleanup Timer	3600 seconds (1 hour)	1200 seconds (20 minutes)	240 seconds (4 minutes)
AAA Validate Clients	Disabled	Disabled	Disabled
Adhoc Reporting	Enabled	Enabled	Enabled
Monitor Mode Report Interval	10 seconds (0:10)	30 seconds (0:30)	60 seconds (1:00)
Minimum RSSI	-128 dBm	-80 dBm	-80 dBm
Transient Interval	600 seconds (10 minutes)	300 seconds (5 minutes)	120 seconds (2 minutes)
Auto Contain **This feature works only on Monitor Mode APs.**	Disabled	Disabled	Disabled
Auto Contain Level	1	1	1
Auto Contain Same SSID	Disabled	Disabled	Disabled
Auto Contain Valid Clients on Rogue AP	Disabled	Disabled	Disabled
Auto Contain Adhoc	Disabled	Disabled	Disabled
Containment Auto Rate	Enabled	Enabled	Enabled

Parameter	Critical	High	Low
Validate Clients with Cisco Connected Mobile Experiences (CMX)	Enabled	Enabled	Enabled
Containment FlexConnect	Enabled	Enabled	Enabled
Rogue Location Discovery Protocol (RLDP)	The device functions as a monitor mode AP if RLDP scheduling is disabled.	The device functions as a monitor mode AP if RLDP scheduling is disabled.	Disabled
Auto Contain RLDP	Disabled	Disabled	Disabled

You can configure all these parameters in the Custom security level.

- A hospital implements the Critical security level to maintain rigorous control over rogue detection with fixed settings.
- A small business chooses the Low security level for straightforward rogue detection with minimal configuration.
- An enterprise IT team uses the Custom security level to tailor all rogue detection parameters to their unique requirements.

Set rogue detection security level (CLI)

Set the wireless rogue detection security level for your network deployment.

Before you begin

Use these steps to set the rogue detection security level.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the rogue detection security level to custom.

Example:

```
Device(config)# wireless wps rogue security-level custom
```

Step 3 Configure the rogue detection security level for small-scale deployments.

Example:

```
Device(config)# wireless wps rogue security-level low
```

Step 4 Configure the rogue detection security level for medium-scale deployments.

Example:

```
Device(config)# wireless wps rogue security-level high
```

Step 5 Configure the rogue detection security level for highly sensitive deployments.

Example:

```
Device(config)# wireless wps rogue security-level critical
```

The device applies the selected rogue detection security level to enhance wireless intrusion protection for your deployment.

Configuring rogue detection (GUI)

Enable rogue access point detection and set parameters in the GUI.

Use rogue detection to identify and manage unauthorized or suspicious APs in your network. Complete this task when you establish or update your wireless security policies.

Before you begin

Use these steps to configure rogue detection using the GUI:

Procedure

Step 1 Choose Configuration, then Tags and Profiles, then AP Join.

Step 2 Click the **AP Join Profile Name** to edit the access point (AP) join profile properties.

Step 3 In the **Edit AP Join Profile** window, click the **Rogue AP** tab.

Step 4 Check the **Rogue Detection** check box to enable rogue detection.

Step 5 In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

Step 6 In the **Rogue Detection Transient Interval** field, enter the interval in seconds (minutes).

Step 7 In the **Rogue Detection Report Interval** field, enter the report interval value in seconds (minutes).

Step 8 In the **Rogue Detection Client Number Threshold** field, enter the threshold for rogue client detection.

Step 9 Check the **Auto Containment on FlexConnect Standalone** check box to enable auto containment.

Step 10 Click **Update and Apply to Device**.

The rogue-detection feature is activated with your configured parameters. This helps you identify and contain unauthorized access points (APs) in the network.

Configure rogue detection (CLI)

You enable and customize rogue detection on Cisco wireless access points using specific CLI commands.

Use these commands to detect and contain unauthorized access points (APs) and improve wireless network security.

Before you begin

Use these steps to configure rogue detection.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Set how often APs consistently scan for rogues after the initial detection.

Example:

```
Device(config)# ap profile profile-name
Device(config)# rogue detection min-transient-time time in seconds
```

You can enter a time between two minutes (120 seconds) and thirty minutes (1,800 seconds). The default is zero.

Note
You can use this feature with any AP mode.

Use the transient interval to control how often APs scan for rogues. APs can also filter rogues based on this value.

This feature provides these advantages:

- Rogue reports from APs to the controller are shorter.
- Transient rogue entries are avoided in the controller.

The system avoids unnecessary memory allocation for transient rogues.

Step 3 Choose a rogue containment option.

Example:

```
Device(config)# ap profile profile-name
Device(config)# rogue detection containment flex-rate
```

The **auto-rate** option contains rogues automatically. The **flex-rate** option contains standalone FlexConnect APs.

Step 4 Turn on rogue detection for all APs.

Example:

```
Device(config)# ap profile profile-name
```

Step 5 Set the interval for rogue reports on monitor mode APs.

Example:

```
Device(config)# ap profile profile-name
Device(config)# rogue detection report-interval time in seconds
```

The valid range for the reporting interval is 10 to 300 seconds.

If the controller detects thousands of rogue APs, the PUBD (Public Utility Bulletin Daemon) process may cause sustained high CPU usage. Increase the Rogue Detection Report Interval to a value higher than the default of 10 to resolve this issue.

Rogue detection is active using your specified parameters on the AP profile. This improves security by monitoring and containing unauthorized devices.

Configure RSSI deviation notification threshold for rogue APs (CLI)

Set the signal strength deviation threshold to trigger notifications for rogue access points on your network.

Before you begin

Use these steps to configure the RSSI deviation notification threshold for rogue APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the RSSI deviation notification threshold for rogue APs.

Example:

```
Device(config)# wireless wps rogue ap notify-rssi-deviation
```

Step 3 Return the system to privileged EXEC mode.

Example:

```
Device(config)# end
```

You can press Ctrl and Z to exit global configuration mode.

The system notifies you when a rogue APs RSSI deviation exceeds the set threshold.

Configure management frame protection (GUI)

Enable and configure management frame protection (MFP) to secure wireless network communications against attacks such as rogue AP impersonation.

Before you begin

Use these steps to configure management frame protection.

Procedure

Step 1 Choose **Configuration**, then **Security**, then **Wireless Protection Policies**.

Step 2 In the **Rogue Policy** tab, under the **MFP Configuration** section, check the **Global MFP State** check box to enable the global MFP state.

Step 3 Check the **AP Impersonation Detection** check box to enable AP impersonation detection.

Step 4 In the **MFP Key Refresh Interval** field, specify the refresh interval in hours.

Step 5 Click **Apply**.

Management frame protection is enabled and configured, providing enhanced security for wireless management frames.

Configure Management Frame Protection (CLI)

Configure Management Frame Protection (MFP) on a device using the command-line interface (CLI).

Before you begin

Use these steps to configure management frame protection.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure management frame protection.

Example:

```
Device(config)# wireless wps mfp
```

Step 3 Configure AP impersonation detection or set the MFP key refresh interval in hours.

Example:

```
Device(config)# wireless wps mfp ap-impersonation
```

```
Device(config)# wireless wps mfp key-refresh-interval
```

key-refresh-interval: Set sthe MFP key refresh interval in hours.

The valid range is one to 24 hours. The default value is 24 hours.

Step 4 Save the configuration, exit configuration mode, and return to privileged **EXEC** mode.

Example:

```
Device(config)# end
```

The device protects wireless management traffic according to the configured settings.

Verify management frame protection

To verify if the Management Frame Protection (MFP) feature is enabled or not, use this command:

```
Device# show wireless wps summary
Client Exclusion Policy
  Excessive 802.11-association failures   : unknown
  Excessive 802.11-authentication failures: unknown
  Excessive 802.1x-authentication         : unknown
  IP-theft                                : unknown
  Excessive Web authentication failure    : unknown
  Failed Qos Policy                       : unknown

Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

To view the MFP details, use this command:

```
Device# show wireless wps mfp summary
Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

Verify rogue events

To verify the rogue event history, run the **show wireless wps rogue ap detailed** command:

```
Device# show wireless wps rogue ap detailed d8b1.901c.3cfd

Rogue Event history

Timestamp                  #Times Class/State Event               Ctx
 RC
--------------------------- ------ ----------- ------------------- ------------------------
 ---
05/01/2020 08:37:03.55645  41616  Mal/CPend   FSM_GOTO            ContPending(NotContYet)
 0x0
05/01/2020 08:37:03.55427  28163  Mal/CPend   EXPIRE_TIMER_START  1200s
 0x0
05/01/2020 08:37:03.55380  28163  Mal/CPend   RECV_REPORT         38ed.18cf.83e0/1
 0x0
05/01/2020 08:36:54.659136 7356   Mal/CPend   NO_OP_UPDATE
 0x0
05/01/2020 08:36:33.347132 3185   Mal/CPend   CHANNEL_CHANGE      e4aa.5d44.fec0/2,36->40
 0x0
05/01/2020 08:25:19.573720 247    Mal/CPend   LRAD_EXPIRE         7c21.0e41.0700/0
 0x0
04/30/2020 07:55:37.977450 2      Mal/CPend   PMF_CONTAINMENT  ContPending(PMFDetected) 0x0
04/30/2020 07:55:37.977242 1      Unc/Alert   INIT_TIMER_DONE     0xab9800439e00024f
 0x0
04/30/2020 07:52:33.600332 1      Unk/Init    INIT_TIMER_START    180s
 0x0
04/30/2020 07:52:33.600326 1      Unk/Init    CREATE
 0x0
```

To verify the impersonations detected due to authentication errors, use this command:

```
Device# show wireless wps rogue ap detailed

Rogue BSSID                              : 0062.ecf3.8d30
Last heard Rogue SSID                    : rogueA
802.11w PMF required                     : No
Is Rogue an impersonator                 : Yes
Is Rogue on Wired Network                : No
Classification                           : Malicious
Manually Contained                       : No
State                                    : Threat
First Time Rogue was Reported            : 01/07/2020 15:51:01
Last Time Rogue was Reported             : 01/08/2020 08:08:35

Number of clients                        : 0

Reported By
  AP Name : AP38ED.18CE.45E0
    MAC Address                          : 38ed.18cf.83e0
    Detecting slot ID                    : 0
    Radio Type                           : dot11g, dot11n - 2.4 GHz
    SSID                                 : rogueA
    Channel                              : 6 (From DS)
    Channel Width                        : 20 MHz
    RSSI                                 : -33 dBm
    SNR                                  : 52 dB
    ShortPreamble                        : Disabled
    Security Policy                      : WPA2/WPA/FT
    Last reported by this AP             : 01/08/2020 08:02:53
    Authentication Failure Count         : 237
```

Verify rogue detection

This section describes the new command for rogue detection.

These commands can be used to verify rogue detection on the device.

Table 18: Verifying Adhoc rogues information

Command	Purpose
show wireless wps rogue adhoc detailed *mac_address*	Displays the detailed information for an Adhoc rogue.
show wireless wps rogue adhoc summary	Displays a list of all Adhoc rogues.

Table 19: Verifying rogue AP information

Command	Purpose
show wireless wps rogue ap clients *mac_address*	Displays the list of all rogue clients associated with a rogue.
show wireless wps rogue ap custom summary	Displays the custom rogue AP information.
show wireless wps rogue ap detailed *mac_address*	Displays the detailed information for a rogue AP.

show wireless wps rogue ap friendly summary	Displays the friendly rogue AP information.
show wireless wps rogue ap list *mac_address*	Displays the list of rogue APs detected by a given AP.
show wireless wps rogue ap malicious summary	Displays the malicious rogue AP information.
show wireless wps rogue ap summary	Displays a list of all Rogue APs.
show wireless wps rogue ap unclassified summary	Displays the unclassified rogue AP information.

Table 20: Verifying Rogue Auto-Containment Information

Command	Purpose
show wireless wps rogue auto-contain	Displays the rogue auto-containment information.

Table 21: Verifying Classification Rule Information

Command	Purpose
show wireless wps rogue rule detailed *rule_name*	Displays the detailed information for a classification rule.
show wireless wps rogue rule summary	Displays the list of all rogue rules.

Table 22: Verifying Rogue Statistics

Command	Purpose
show wireless wps rogue stats	Displays the rogue statistics.

Table 23: Verifying Rogue Client Information

Command	Purpose
show wireless wps rogue client detailed *mac_address*	Displays detailed information for a Rogue client.
show wireless wps rogue client summary	Displays a list of all the Rogue clients.

Table 24: Verifying Rogue Ignore List

Command	Purpose
show wireless wps rogue ignore-list	Displays the rogue ignore list.

Examples: rogue detection onfiguration

This example shows how to configure the minimum RSSI that a detected rogue AP needs to be at, to have an entry created in the device:

```
Device# wireless wps rogue ap notify-min-rssi 100
```

This example shows how to configure the classification interval:

```
Device# configure terminal
Device(config)#
Device(config)#
Device(config)# end
Device# show wireless wps rogue client /show wireless wps rogue ap summary
```

Configure rogue policies (GUI)

Use this task to define and customize rogue wireless protection policies. These policies help the system detect and respond to unauthorized wireless activity.

Before you begin

Perform the steps in this section to configure rogue policies.

Procedure

Step 1 Choose **Configuration**, then **Security**, then **Wireless Protection Policies**.

Step 2 In the **Rogue Policies** tab, select the security level from the **Rogue Detection Security Level** drop-down.

Step 3 In the **Expiration timeout for Rogue APs** field, enter the timeout value in seconds.

Step 4 Select the **Validate Rogue Clients against AAA** check box to validate rogue clients using the AAA server.

Step 5 Select the **Validate Rogue APs against AAA** check box to validate rogue access points using the AAA server.

Step 6 In the **Rogue Polling Interval** field, enter the interval in seconds at which the system polls the AAA server for rogue information.

Step 7 Select the **Detect and Report Adhoc Networks** check box to enable detection of rogue ad hoc networks.

Step 8 In the **Rogue Detection Client Number Threshold** field, enter the number of clients at which the system generates an SNMP trap.

Step 9 In the **Auto Contain** section, enter these details.

Step 10 Select the containment level from the **Auto Containment Level** drop-down.

Step 11 Select the **Auto Containment only for Monitor Mode APs** check box to limit automatic containment to monitor-mode APs.

Step 12 Select the **Using our SSID** check box to limit automatic containment to rogue APs that use an SSID that is configured on the controller.

Step 13 Select the **Adhoc Rogue AP** check box to enable automatic containment for ad hoc rogue APs.

Step 14 Click **Apply**.

The system updates rogue policies to enhance detection and containment of unauthorized wireless threats according to your configuration.

Configure rogue policies (CLI)

Establish rogue policies to enhance network security by managing rogue access points and clients.

This configuration is essential in environments where rogue devices may pose a security threat to the network.

Before you begin

Ensure you have access to the device's global configuration mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the expiration time for rogue entries.

Example:

```
Device(config)# wireless wps rogue ap timeout timout-in-seconds
```

Valid range for the time in seconds is 240 seconds to 3600 seconds.

Step 3 Configure the minimum RSSI notification threshold for rogue clients.

Example:

```
Device(config)# wireless wps rogue client notify-min-rssi RSSI-threshold
```

Valid range for the RSSI threshold in dB is -128 dB to -70 dB.

Step 4 Configure the RSSI deviation notification threshold for rogue clients.

Example:

```
Device(config)# wireless wps rogue client notify-min-deviation RSSI-threshold-value
```

Valid range for the RSSI threshold in dB is 0 dB to 10 dB.

Step 5 Configure rogue AP AAA validation interval.

Example:

```
Device(config)# wireless wps rogue ap aaa polling-interval AP-AAA-interval-value
```

The valid range for the AP AAA interval in seconds is 60 seconds to 86400 seconds.

Step 6 Enable detecting and reporting adhoc rogue (IBSS).

Example:

```
Device(config)# wireless wps rogue adhoc
```

Enables detecting and reporting adhoc rogue (IBSS).

Step 7 Configure the rogue client per a rogue AP SNMP trap threshold.

Example:

```
Device(config)# wireless wps rogue client client-threshold threshold-value
```

The valid range for the threshold is 0 to 256.

The rogue policies are now configured, enhancing the network's security against rogue devices.

Rogue Location Discovery Protocol (RLDP)

A rogue location discovery protocol is a network security protocol that

- detects rogue access points by actively investigating their presence
- enables authorized access points to associate with rogue devices and communicate findings to network controllers, and
- enhances wireless network protection by identifying open and NATed rogue APs.

How RLDP Works

Rogue Location Discovery Protocol (RLDP) operates when a rogue access point uses open authentication (lacks password protection). RLDP instructs an authorized AP to temporarily disconnect its clients, switch to the rogue channel, and attempt to join the rogue AP as a client.

Once associated, the AP requests an IP address from the rogue AP and sends a UDP packet (port 6352) containing connection details to the controller. If the controller receives this packet, it triggers an alarm to alert administrators about the presence of a rogue AP on the wired network.

RLDP offers high accuracy in rogue AP detection, especially for open and NAT-enabled rogue devices. However, RLDP packets may be blocked if filtering rules exist between the controller network and the rogue network.

Note The RLDP packets are unable to reach the embedded wireless controller if filtering rules are placed between the embedded wireless controller 's network and the network where the rogue device is located.

Guidelines to manage RLDP

- Use Rogue Location Discovery Protocol (RLDP) to detect rogue access points configured for open-authentication.
- Use RLDP to detect rogue access points that broadcast a Basic Service Set Identifier (BSSID). In this case, the access point broadcasts its Service Set Identifier in beacons.
- You can use RLDP to detect only rogue access points on the same network. RLDP does not function if an access list in the network blocks RLDP traffic from the rogue access point to the embedded wireless controller.
- You cannot use RLDP on 5-GHz (5-gigahertz) Dynamic Frequency Selection (DFS) channels.
- If you enable RLDP on mesh APs and they perform RLDP tasks, the mesh APs will be dissociated from the embedded wireless controller. To prevent this, disable RLDP on mesh APs.

- If you enable RLDP on non-monitor APs, clients may experience connectivity outages while RLDP runs.

Best practices for RLDP initiation commands and configuration

Your embedded wireless controller collects information about rogue access points and clients from nearby APs.

When your embedded wireless controller discovers a rogue AP, it uses Rogue Location Discovery Protocol (RLDP) (if enabled and the rogue detector mode AP is connected) to determine whether the rogue is attached to your network.

The Embedded Wireless Controller initiates RLDP on rogue devices that use open .

If RLDP uses FlexConnect or local mode access points, clients disconnect briefly and reconnect after RLDP completes.

When a rogue access point appears , RLDP starts automatically.

You can configure the embedded wireless controller to use RLDP on all access points or only on those set to monitor (listen-only) mode.

Monitoring-mode access points let you detect rogue access points automatically in crowded RF spaces without creating interference or affecting data access point functionality.

If both monitor and local (data) access points are nearby, the embedded wireless controller selects the monitor access point for RLDP.

If RLDP determines that the rogue is on your network, choose to contain the rogue manually or automatically.

RLDP checks whether rogue access points that use open authentication are present on the wire only once by default.

You can configure the number of retries using the .

Use these best practices to initiate and configure RLDP on the embedded wireless controller.

1. Initiate RLDP manually from the controller CLI.
2. Schedule RLDP from the controller CLI.
3. Configure auto RLDP on your controller from the CLI or GUI after setting the correct security level.

Additional guidance

- You can configure auto RLDP only when the rogue detection security level is set to custom.
- Enable either auto RLDP or scheduled RLDP, but not both at the same time.

Restrictions for RLDP

- You can use Rogue Location Discovery Protocol (RLDP) only with open rogue APs that broadcast their SSID with authentication and encryption disabled.
- Ensure the managed AP acting as a client can obtain an IP address using DHCP on the rogue network.
- You can use manual RLDP to run an RLDP trace on a rogue device multiple times.

- During the RLDP process, the AP cannot serve clients, which reduces performance and connectivity for local mode APs. To prevent this issue, enable RLDP only on Monitor Mode APs.
- RLDP does not connect to rogue APs operating in 5GHz DFS channels.
- You can use RLDP only on Cisco IOS APs.

Rogue Location Discovery Protocol (RLDP) process

Summary

Use Rogue Location Discovery Protocol (RLDP) to help Unified APs identify and verify whether a rogue device is connected to the network.

RLDP associates with the rogue device, obtains network parameters, and reports the results to controllers.

Workflow

1. Identify the closest Unified AP to the rogue using signal strength values.
2. The AP then connects to the rogue as a WLAN client, attempting three associations before timing out.
3. If association is successful, the AP then uses DHCP to obtain an IP address.
4. If an IP address is obtained, the AP (acting as a WLAN client) sends a UDP packet to each of the embedded wireless controller's IP addresses.
5. If the embedded wireless controller receives even one RLDP packet from the client, the system marks the rogue as on-wire.

Result

You can use RLDP to automatically detect and confirm rogue devices connected to your enterprise wired network.

RLDP supports rapid security response.

Configuring RLDP for Generating Alarms (GUI)

Procedure

Step 1 Choose **Configuration > Security > Wireless Protection Policies**.

Step 2 In the **RLDP** tab, use the **Rogue Location Discovery Protocol** drop-down to select one of the following options:

a) **Disable**: Disables RLDP on all the access points. **Disable** is the default option.
b) **All APs**: Enables RLDP on all APs.
c) **Monitor Mode APs**: Enables RLDP only on APs in the monitor mode.

Note
The **Schedule RLDP** check box is enabled only if the **Disable** option is selected. The Schedule RLDP check box remains disabled when you select the **All APs** option or the **Monitor Mode APs** option.

Step 3 In the **Retry Count** field, specify the number of retries that should be attempted. The range allowed is between 1 and 5.

Step 4 Click **Apply**.

Configure RLDP to generate alarms (CLI)

Enable the Rogue Location Discovery Protocol (RLDP) to generate alarms using CLI commands.

Before you begin

Follow these steps to configure RLDP to create alarms.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable RLDP to generate alarms.

Example:

```
Device(config)# wireless wps rogue ap rldp alarm-only
```

```
Device(config)# wireless wps rogue ap rldp alarm-only monitor-ap-only
```

In this method, RLDP is always enabled.

The **monitor-ap-only** keyword is optional.

The command with only the **alarm-only** keyword enables RLDP without restrictions on AP mode.

The command with the "alarm-only monitor-ap-only" keyword enables RLDP only in monitor-mode access points.

Step 3 Go to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can press **Ctrl-Z** to exit global configuration mode.

RLDP generates alarms when the system detects rogue access points, according to your configuration.

Configure a schedule for RLDP (GUI)

Define and apply a scheduled operation for the Rogue Location Discovery Protocol (RLDP) to automate rogue device detection on your wireless network.

Before you begin

Use these steps to configure a schedule for RLDP.

Procedure

Step 1 Choose **Configuration > Security > Wireless Protection Policies**.

Step 2 In the **RLDP** tab, choose these options from the **Rogue Location Discovery Protocol** drop-down list.

Disable (default): Disables RLDP on all access points.

Step 3 In the **Retry Count** field, enter a value between 1 and 5 for the number of retries.

Step 4 Select the **Schedule RLDP** check box, and then specify the days, start time (for example, 8:00 a.m. [0800]), and end time (for example, 5:00 p.m. [1700]) for the process.

Step 5 Click **Apply**.

The system schedules RLDP based on your configuration and automatically detects rogue APs during the specified periods.

Configure a schedule for RLDP (CLI)

Set up a scheduled Rogue Location Discovery Protocol (RLDP) scan to run at specific days and times using CLI.

Before you begin

Use these steps to configure an RLDP schedule using CLI.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable Rogue Location Discovery Protocol (RLDP) based on a scheduled day, start time, and end time.

Example:

```
Device(config)# wireless wps rogue ap rldp schedule day Monday start start-time end end-time
```

day is the day when the RLDP scheduling can be done. The values are Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday.

start-time is the start time for scheduling RLDP for the day. You need to enter start time in **HH:MM:SS** format.

end-time is the end time for scheduling RLDP for the day. You need to enter end time in **HH:MM:SS** format.

Step 3 Enable the schedule.

Example:

```
Device(config)# wireless wps rogue ap rldp schedule
```

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RLDP schedule runs automatically on the specified days and times, enabling periodic rogue AP detection.

Configure an RLDP for auto-contain (GUI)

Enable RLDP-based automatic containment of rogue devices detected on the wired network.

Before you begin

Use these steps to configure RLDP for auto-contain.

Procedure

Step 1 Choose **Configuration > Security > Wireless Protection Policies** .

Step 2 In the **Rogue Policies** tab, under the **Auto Contain** section, check the **Rogue on Wire** checkbox.

Step 3 Click **Apply** .

The system will automatically contain rogue devices detected through RLDP on the wired network.

Configure an RLDP for auto-contain (CLI)

Enable Rogue Location Discovery Protocol (RLDP) auto-contain mode on a wireless LAN controller using CLI.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable RLDP to perform auto-contain. This method always keeps RLDP enabled.

Example:

```
Device config)# wireless wps rogue ap rldp auto-contain
```

```
Device (config)# wireless wps rogue ap rldp auto-contain monitor-ap-only
```

The **monitor-ap-only** keyword is optional.

Using the command with only **auto-contain** keyword enables RLDP without any restriction on the AP mode.

The command with **auto-contain <monitor-ap-only>** keyword enables RLDP in monitor mode access points only.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

RLDP is enabled for auto-contain. Rogue APs matching the configured behavior are automatically contained by the controller.

Configure RLDP retry times on rogue APs (GUI)

Set the number of RLDP retry attempts on rogue wireless APs.

Before you begin

Use these steps to configure RLDP retry times for rogue APs.

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies** .

Step 2 On the **Wireless Protection Policies** page, click the **RLDP** tab.

Step 3 Enter the RLDP retry attempt value for rogue access points in the **Retry Count** field.

The valid range is between 1 to 5.

Step 4 Save the configuration.

The RLDP retry count value for rogue APs are updated and applied on the network controller.

Configure RLDP retry times on rogue APs (CLI)

Set the number of RLDP (Rogue Location Discovery Protocol) retry attempts for rogue APs.

Before you begin

Use these steps to configure RLDP retry times on rogue APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable RLDP retry times on rogue APs.

Example:

```
Device(config)# wireless wps rogue ap rldp retries num-entries
```

Here, *num-entries* is the number of RLDP retry times for each of the rogue APs.

The valid range is 1 to 5.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The device updates the RLDP retry times for rogue AP as specified, affecting future rogue AP RLDP attempts.

Verify rogue AP RLDP

These commands can be used to verify rogue AP RLDP.

Table 25: Verify rogue AP information

Command	Purpose
show wireless wps rogue ap rldp detailed *mac_address*	Displays the RLDP details for a rogue AP.
show wireless wps rogue ap rldp in progress	Displays the list of in-progress RLDP.
show wireless wps rogue ap rldp summary	Displays the summary of RLDP scheduling information.

Wireless Service Assurance (WSA) rogue events

Wireless Service Assurance (WSA) rogue events are telemetry notifications that replicate the information of corresponding SNMP traps. Support is available in Release 16.12.x, where x denotes a release version.

- You receive details such as the MAC address of the rogue AP.
- You receive information about the managed AP and the radio that detected the rogue AP with the strongest RSSI.
- You receive event-specific data, such as SSID; channel for potential honeypot events; and MAC address of the impersonating AP for impersonation events.

WSA Rogue Events: Details and Support

For all exported events, these details are provided to the wireless service assurance (WSA) infrastructure:

- MAC address of the rogue AP
- Details of the managed AP and the radio that detected the rogue AP with the strongest RSSI

- Event-specific data such as SSID; channel for potential honeypot events; and MAC address of the impersonating AP for impersonation events.

You can scale the WSA rogue events feature up to four times the maximum number of supported access points (APs). You can also scale it to one-half of the maximum number of supported clients.

The WSA rogue events feature is supported on Cisco Catalyst Center and other third-party infrastructure.

Wireless service assurance rogue events

Configure your wireless device to send service assurance rogue event data to the event queue.

Before you begin

To configure wireless service assurance for rogue events, complete these steps.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable wireless service assurance.

Example:

```
Device# network-assurance enable
```

Step 3 Enable wireless service assurance for rogue devices.

Example:

```
Device# wireless wps rogue network-assurance enable
```

This ensures that the wireless service assurance (WSA) rogue events are sent to the event queue.

Your device sends wireless service assurance rogue events to the event queue for enhanced monitoring.

Monitor wireless service assurance rogue events

View wireless service assurance (WSA) rogue event statistics and details.

Before you begin

Use these steps to monitor wireless service assurance rogue events.

Procedure

Step 1 **show wireless wps rogue stats**

Example:

```
Device# show wireless wps rogue stats
            WSA Events
            Total WSA Events Triggered          : 9
            ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 2
            ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 3
            ROGUE_AP_IMPERSONATION_DETECTED     : 4
            Total WSA Events Enqueued           : 6
            ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 1
            ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 2
            ROGUE_AP_IMPERSONATION_DETECTED     : 3
```

In this example, nine events occurred, but only six events were enqueued. Three events occurred before you enabled the WSA rogue feature.

Step 2 **show wireless wps rogue stats internal**

show wireless wps rogue ap detailed *rogue-ap-mac-addr*

These commands display information about WSA events in the event history.

You can track WSA rogue event activity, investigate event history, and verify system responsiveness to rogue threats.

Rogue PMF containment

Rogue PMF containment is a wireless security feature that

- uses 802.11w Protected Management Frames (PMF) to contain rogue APs and clients
- operates on centrally switched WLANs when the radio channel of the detecting AP matches the rogue AP's channel, and
- activates only when certain mode and channel conditions are met to secure the network against unauthorized devices.

Feature history

Table 26: Feature history table for rogue PMF containment

Feature name	Release information	Feature description
Rogue PMF containment	Cisco IOS XE 17.12.1	Starting with Cisco IOS XE Dublin 17.12.1, the controller contains a rogue AP with 802.11w Protected Management Frame (PMF) on centrally switched wireless LANs. Containment occurs if the client-serving radio channel of a rogue-detecting AP matches the channel of the corresponding rogue AP.

Operational scenarios

PMF containment occurs in these scenarios:

- You can use PMF containment only in the local mode.
- You can perform PMF containment only for rogue clients that have not joined a rogue AP.
- You can use PMF containment only if a rogue-detecting AP shares the same primary channel with a rogue client.
- You cannot use PMF containment on DFS channels, even if a DFS channel serves as the client-serving channel.
- PMF containment works only if at least one WLAN operates on the serving radio.

For information about APs that support the Rogue PMF Containment feature, see Cisco AP Feature Matrix.

Enable rogue PMF containment

Enable PMF containment to protect your wireless network from rogue APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile ap-profile
```

Step 3 Enable PMF-denial rogue AP containment.

Example:

```
Device(config-ap-profile)# rogue detection containment pmf-denial
```

Step 4 Enable PMF-denial type deauthentication rogue AP containment.

Example:

```
Device(config-pmf-denial)# pmf-deauth
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-ap-profile)# end
```

Rogue AP PMF containment is enabled for the specified AP profile.

```
Device# configure terminal
Device(config)# ap profile pmf-ap-profile
```

```
Device(config-ap-profile)# rogue detection containment pmf-denial
Device(config-pmf-denial)# pmf-deauth
Device(config-ap-profile)# end
```

Verify PMF containment

To verify PMF containment and the relevant statistics, use these commands.

To view the summary of containment details for all AP radios, use this command

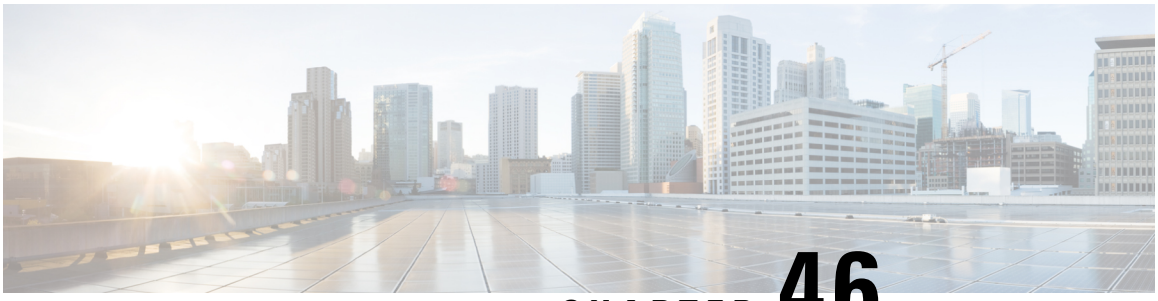
```
Device# show wireless wps rogue containment summary

Rogue Containment activities for each managed AP

AP: 687d.b45f.2ae0  Slot: 1
  Active Containments   : 3
   Containment Mode     : DEAUTH_PMF
   Rogue AP MAC         : 687d.b45f.2a2d
   Containment Channels : 40
```

To verify the rogue statistics, use this command:

```
Device# show wireless wps rogue stats
.
.
.
 States
  Alert                          : 256
  Internal                       : 0
  External                       : 0
  Contained                      : 1
  Containment-pending            : 0
  Threat                         : 0
  Pending                        : 0
Rogue Clients
  Total/Max Scale                : 20/16000
  Contained                      : 0
  Containment-pending            : 0
.
.
.
```



CHAPTER 46

Classifying Rogue Access Points

- Rogue AP classifications, on page 453

Rogue AP classifications

A rogue AP classification is a wireless security mechanism that

- enables administrators to group rogue APs as Friendly, Malicious, , or Unclassified,
- uses configurable rules to automate the categorization and state assignment of unauthorized APs, and
- allows both automatic and manual reclassification of rogues based on network security requirements.

Additional reference information

- By default, no classification rules are active, so all unknown access points are assigned the Unclassified state. Administrators must enable classification rules to begin organizing rogue access points.
- When rules are enabled, all rogue access points in the Alert state are reclassified automatically based on the latest conditions and rule configurations.
- Changing or adding a rule triggers reclassification of all applicable rogue access points that are in the Alert state.
- You can manually move any rogue or ad hoc rogue access point to Unclassified and state, which represents the default state. Manually moved rogues are subject to rule-based reclassification.
- Rule-based classification does not apply to ad hoc rogues and rogue clients.
- You can configure up to 64 rogue classification rules per embedded wireless controller .
-

How the controller classifies rogue access points

When the controller software receives a rogue report from one of its managed access points, it follows this process:

- If the unknown access point is listed in the friendly MAC address list, the embedded wireless controller classifies it as Friendly.

- If it is not on the friendly MAC list, the embedded wireless controller applies the configured rogue classification rules.
-
- If a rogue matches the criteria in a classification rule, the embedded wireless controller assigns the classification type specified in that rule.
- If a rogue does not match any configured rules, it remains Unclassified.
- If a rogue access point is detected on the same wired network, the embedded wireless controller marks its state as Threat and classifies it as Malicious automatically (regardless of rules). You can manually contain such rogues, changing their state to Contained. If the rogue becomes unavailable, the embedded wireless controller moves it to the Alert state until you manually contain it.
- Administrators can always manually change the classification type and state of an access point as needed.

Examples: Classification mapping

Rule-Based Classification Type	Rogue State
Friendly	• Internal—If the unknown AP poses no threat to WLAN security, you can manually configure it as Friendly, Internal. An example of this would be the APs in your lab network. • External—If the unknown AP is outside the network and poses no threat to WLAN security, you can manually configure it as Friendly, External. An example of this would be the AP in your neighboring coffee shop. • Alert—
Malicious	• Alert— • Threat—The unknown AP is found to be on the network and poses a threat to WLAN security. • Contained—The unknown AP is contained.
Unclassified	• Alert— • Contained—The unknown AP is contained.

As mentioned earlier, the embedded wireless controller can automatically change the classification type and rogue state of an unknown AP based on user-defined rules. Alternatively, you can manually move the unknown AP to a different classification type and rogue state.

Allowable classification type and rogue state transitions

Assigning rogue APs classifications is like sorting emails automatically into spam or inbox based on filters (rules), while still allowing the user to manually move an email to a different folder when needed.

Rogue containment methods

A rogue containment method is a wireless security approach that

- detects unauthorized access points in the network environment,
- applies different containment strategies including manual, rule-based, and auto-containment, and
- prioritizes and restricts the actions of access points based on containment configuration and AP operating mode.

Containment methods and priorities

- **Manual rogue containment (Priority 1):** Administrators manually designate which rogue APs to contain, offering the highest control and flexibility. Containment levels can range from 1 to 4 APs participating.
- **Rule-based rogue containment (Priority 2):** The controller automatically contains rogue APs based on predefined rules, always using 1 AP for containment.
- **Auto-containment (Priority 3):** The controller automatically contains rogue APs, but only when specific options (such as "Using our SSID", "A valid client on the Rogue AP", or "Ad-hoc Rogue AP") are enabled.

Auto-containment for monitor mode APs

- When you enable 'Auto Containment only for Monitor Mode APs', only monitor mode APs perform auto-containment. Local or FlexConnect mode APs do not.
- This option applies only to auto-containment. It does not affect your manual or rule-based methods.
- Other AP modes (local or FlexConnect) still enforce rogue containment actions outside the scope of this auto-containment setting.

Containment prioritization and exceptions

- Manual containment always takes highest priority, followed by rule-based, with auto-containment last.
- If a rogue AP matches both manual or rule-based configurations, auto-containment does not occur for that AP.

Example

An ad-hoc rogue AP detected with a valid client connected triggers auto-containment by monitor mode APs only if the corresponding option is enabled. If manual containment is also configured for this rogue AP, manual containment takes precedence and disables auto-containment for that case.

Guidelines and restrictions for classifying Rogue APs

Classifying Custom type rogues is tied to rogue rules. Therefore, it is not possible to manually classify a rogue as Custom. Custom class change can occur only when rogue rules are used.

Some are sent for containment by rule and every 30 minutes for rogue classification change.

Rogue rules are applied on every incoming new rogue report in the embedded wireless controller in the order of their priority.

After a rogue satisfies a rule and is classified, it does not move down the priority list for the same report.

Previously classified rogue gets re-classified on every new rogue report with the following restrictions:

- Rogues which are classified as friendly by rule and whose state is set to ALERT, go through re-classification on receiving the new rogue report.
- If a rogue is classified as friendly by the administrator manually, then the state is INTERNAL and it does not get re-classified on successive rogue reports.
- If rogue is classified as malicious, irrespective of the state it does not get re-classified on subsequent rogue reports.

If a rogue AP is classified as friendly

Until the controller discovers all the APs through neighbor reports from APs, the rogue APs are kept in unconfigured state for three minutes after they are detected. After 3 minutes, the rogue policy is applied on the rogue APs and the APs are moved to unclassified, friendly, malicious, or custom class. Rogue APs kept in unconfigured state means that no rogue policy has yet been applied on them.

When a rogue BSSID is submitted for a containment on Cisco Catalyst 9800 Series Wireless Controller, if the controller has enough resources, it will contain. The APs that detect the particular contained rogue AP starts broadcasting the DEAUTH packets.

Wireless client connected to the contained rogue BSSID will disconnect once DEAUTH packets are received. However, when the client assumes being in a connected state, repeatedly tries to reconnect and the wireless client's user browsing experience would be badly affected.

Also, in a high RF environment like that of a stadium, though DEAUTH packets are broadcasted, client does not receive all of them because of RF disturbance. In this scenario, the client may not be fully disconnected but will be affected badly.

The rouge AP manual classification limit has been enhanced from 625 to 10,000 configurations at a time. The rouge client manual classification limit has been enhanced from 625 to 10,000 configurations at a time.

How to classify rogue APs

-

Classify a rogue AP manually (GUI)

Set the correct classification for rogue access points to maintain wireless network security and compliance.

Use this task after the system detects unclassified APs and you need to determine their status.

Procedure

Step 1 Choose **Monitoring > Wireless** > **Rogues**.

Step 2 In the **Unclassified** tab, select an AP to view the detail in the lower pane.

Step 3 Use the **Class Type** drop-down to set the status.

Step 4 Click **Apply**.

The AP is reclassified and removed from the unclassified list; its status is updated accordingly.

Classify rogue APs and clients manually (CLI)

Classify rogue access points and clients to enhance network security.

This procedure is used in environments where rogue access points and clients need to be identified and managed to prevent unauthorized access to the network.

Before you begin

Ensure you have the necessary permissions to configure the device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Detect and report the ad hoc rogue.

Example:

```
Device(config)# wireless wps rogue adhoc alert mac-addr
```

Enter one of these options after you enter the **adhoc** keyword:

- **alert**—Sets the ad hoc rogue access point to alert mode. If you choose this option, enter the MAC address for the *mac-addr* parameter.
- **auto-contain**—Sets the automatically containing ad hoc rogue to auto-contain mode.
- **contain**—Sets the containing ad hoc rogue access point to contain mode. If you choose this option, enter the MAC address for the *mac-addr* parameter and containment level for the *containment-level* parameter. The valid range for *containment-level* is from 1 to 4.
- **external**—Sets the ad hoc rogue access point as **external**. If you choose this option, enter the MAC address for the *mac-addr* parameter.
- **internal**—Sets the ad hoc rogue access point as **internal**. If you choose this option, enter the MAC address for the *mac-addr* parameter.

Step 3 Configure the rogue APs.

Example:

```
Device(config)# wireless wps rogue ap malicious mac-addr state contain containment-level
```

Example:

```
Device(config)# wireless wps rogue ap malicious 74a0.2f45.c520 state contain 3
```

Enter one of the following options after the **ap** keyword:

- **friendly**—Configures the friendly rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **internal** or **external**.
- **malicious**—Configures the malicious rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **alert** or **contain**.
- **alert**—Sets the malicious rogue access point to **alert** mode.
- **contain**—Sets the malicious rogue access point to **contain** mode. If you choose this option, enter the containment level for the *containment-level* parameter. The valid range is from 1 to 4.

Step 4 Configure the rogue clients.

Example:

```
Device(config)# wireless wps rogue client contain mac-addr containment-level
```

Example:

```
Device(config)# wireless wps rogue client contain 74a0.2f45.c520 2
```

Enter the following option after you enter the **client** keyword:

contain—Contains the rogue client. After you choose this option, enter the MAC address for the *mac-addr* parameter and the containment level for *containment-level* parameter. The valid range for *containment-level* is from 1 to 4.

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure rogue classification rules (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 In the **Wireless Protection Policies** page, choose **Rogue AP Rules** tab.

Step 3 On the **Rogue AP Rules** page, click the name of the **Rule** or click **Add** to create a new one.

Step 4 In the **Add/Edit Rogue AP Rule** window that is displayed, enter the name of the rule in the **Rule Name** field.

Step 5 Choose the rule type from the following **Rule Type** drop-down list options:

- Friendly
- Malicious
- Unclassified

- Custom

Configure rogue classification rules (CLI)

Establish rules for classifying rogue access points to enhance network security.

This configuration is used in environments where rogue access points need to be identified and classified to prevent unauthorized access.

Before you begin

Ensure you have the necessary permissions to configure rogue classification rules on the device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create or enable a rule and set its priority.

Example:

```
Device(config)# wireless wps rogue rule rule-name priority priority
```

When creating a rule, you must enter the priority for the rule.

Note
After creating a rule, you can edit the rule and change the priority only for the rogue rules that are disabled. You cannot change the priority for the rogue rules that are enabled. While editing, changing the priority for a rogue rule is optional.

Step 3 Classify a rule.

Example:

```
Device(config)# wireless wps rogue rule rule_3 priority 3
Device(config-rule)# classify friendly state {alert | external | internal}
```

- **friendly**—Configures the friendly rogue access points. After that enter the **state** keyword followed by either of these options: **alert**, **internal**, or **external**. If you select an **internal** option, it indicates that you trust a foreign access point. If you select an **external** option, it indicates that you acknowledge the presence of a rogue access point.
- **malicious**—Configures the malicious rogue access points. After that enter the state keyword followed by either of these options: alert or contained.
- **alert**—Sets the malicious rogue access point to alert mode.
- **contained**—Sets the malicious rogue access point to contained mode.

Step 4 Add conditions to a rule.

Example:

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# condition {client-count condition_value | duration duration_value |
encryption | infrastructure | rssi | ssid ssid_name | wildcard-ssid}
```

Adds the following conditions to a rule, which the rogue access point must meet:

- **client-count**—Requires that a minimum number of clients be associated to the rogue access point. For example, if the number of clients associated to the rogue access point is greater than or equal to the configured value, the access point could be classified as Malicious. If you choose this option, enter the minimum number of clients to be associated to the rogue access point for the parameter. The valid range is from 1 to 10 (inclusive), and the default value is 0.
- **duration**—Requires that the rogue access point be detected for a minimum period of time. If you choose this option, enter a value for the minimum detection period for the parameter. The valid range is from 0 to 3600 seconds (inclusive), and the default value is 0 seconds.
- **encryption**—Requires that the advertised WLAN does not have encryption enabled. You can choose **any** for any type of encryption, **off** for no encryption, **wpa1** for WPA encryption, **wpa2** for WPA2 encryption, **wpa3-owe** for WPA3 OWE encryption, or **wpa3-sae** for WPA3 SAE encryption.
- **infrastructure**—Requires the SSID to be known to the controller.
- **rssi**—The valid range is from –95 to –50 dBm (inclusive).
- **ssid**—Requires the rogue access point to have a specific SSID. You could specify up to 25 different SSIDs. You should specify an SSID that is not managed by the controller. If you choose this option, enter the SSID for the parameter.
- **wildcard-ssid**—Allows you to specify an expression that could match an SSID string. You can specify up to 25 of these SSIDs.

Step 5 Specify matching conditions for the rule.

Example:

```
Device(config)# wireless wps rogue rule rule_3 priority 3
Device(config-rule)# match {all | any}
```

Specifies whether a detected rogue access point must meet all or any of the conditions specified by the rule for the rule to be matched and the rogue access point to adopt the classification type of the rule.

Step 6 Set a command to its default.

Example:

```
Device(config-rule)# default
```

Step 7 Exit the sub-mode.

Example:

```
Device(config)# wireless wps rogue rule rule_3 priority 3
Device(config-rule)# exit
Device(config)#
```

Step 8 Disable a particular rogue rule.

Example:

```
Device(config)# wireless wps rogue rule rule_3 priority 3
Device(config-rule)# shutdown
```

Disables a particular rogue rule. In this example, the rule *rule_3* is disabled.

Step 9 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Step 10 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 11 Disable all the rogue rules.

Example:

```
Device(config)# wireless wps rogue rule shutdown
```

Step 12 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

You have now configured the rogue classification rules, enabling effective management of rogue access points in the network..

Monitor rogue classification rules

You can monitor the rogue classification rules using these commands.

Table 27: Commands for monitoring rogue classification rules

Command	Purpose
show wireless wps rogue rule detailed	Displays detailed information of a classification rule.
show wireless wps rogue rule summary	Displays a summary of the classification rules.

Examples: classifying rogue AP

This example shows how to classify a rogue AP with MAC address 00:11:22:33:44:55 as malicious and mark it for being contained by 2 managed APs:

```
Device# configure terminal
Device(config)# wireless wps rogue ap malicious 0011.2233.4455 state contain 2
```

This example shows how to create a rule that can categorize a rogue AP that is using SSID **my-friendly-ssid**, and it is seen for at least for 1000 seconds as friendly internal:

```
Device# configure terminal
```

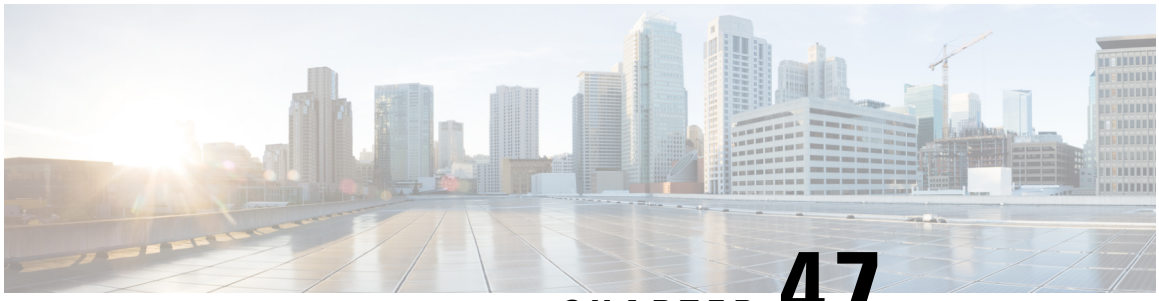
```
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition ssid my-friendly-ssid
Device(config-rule)# condition duration 1000
Device(config-rule)# match all
Device(config-rule)# classify friendly state internal
Device(config-rule)# no shutdown
```

This example shows how to apply a condition that a rogue access point must meet:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition client-count 5
Device(config-rule)# condition duration 1000
Device(config-rule)# no shutdown
Device(config-rule)# end
```

This example shows a condition to classify rogue devices with the controller SSIDs as malicious:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)#  classify malicious state alert
Device(config-rule)#  condition duration 30
Device(config-rule)#  condition infrastructure ssid
Device(config-rule)#  match all
Device(config-rule)#  no shutdown
Device(config-rule)# end
```



CHAPTER 47

Configure Secure Shell

Note Starting with, Secure Shell Version 1 (SSHv1) is deprecated.

• SSH, on page 463

SSH

A secure shell (SSH) protocol is a network security mechanism that

- establishes encrypted and authenticated connections between remote devices. and
- protects data confidentiality and integrity during communication.
- supports multiple version standards, including SSHv1 and SSHv2.

How SSH sorks and security benefits

Secure Shell (SSH) protocols offer a secure alternative to older remote access methods, such as Telnet, by encrypting data exchanged between a client and a server. Authentication further ensures that only authorized users can access devices. SSH increases security for network administrators by preventing eavesdropping and protecting credentials.

- Connecting securely to a router or switch for remote configuration.
- Transferring files securely between devices using SCP, which operates over SSH.

SSH and device access

A Secure Shell (SSH) protocol is a network access method that

- uses strong encryption and authentication to secure remote connections
- offers enhanced security over Telnet by encrypting session and authentication data, and
- supports both IPv4 and IPv6 environments for flexible device access.

Additional information

SSH is a secure protocol for remote device access, supporting both IPv4 and IPv6, and provides strong encryption for authenticated sessions.

SSH servers and integrated clients

A SSH integrated client is a feature that - runs over the SSH protocol to provide device authentication - enables secure, encrypted connections to other devices, and - supports various user authentication methods.

Supported versions and functionalities

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 switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

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

General 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.

SCP

A Secure Copy Protocol is a method for securely transferring files that

- provides a secure and authenticated method for copying switch configurations or switch image files
- relies on Secure Shell (SSH) for secure transport, and
- requires correct configuration of SSH, authentication, and authorization on the switch.

Supporting reference information

For SSH to work, the switch needs an RSA public or 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 a 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

A Secure Copy Protocol is a network protocol that

- provides a secure and authenticated method for copying files
- relies on SSH for security, and
- requires authentication, authorization, and accounting (AAA) authorization.

Additional information on SCP

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite.

To configure the SCP, you should understand the SCP concepts.

SFTP support

SFTP client support is introduced from Cisco IOS XE Gibraltar 16.10.1 release onwards. SFTP client is enabled by default and no separate configuration required. The SFTP procedures can be invoked using the **copy** command, which is similar to that of **scp** and **tftp** commands. A typical file download procedure using **sftp** command can be carried out as shown below:

copy sftp://*user* :*password @server-ip*/*file-name* **flash0://** *file-name*

For more details on the **copy** command, see the following URL:
https://www.cisco.com/c/m/en_us/techdoc/dc/reference/cli/nxos/commands/fund/copy.html

Prerequisites for configuring 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 SSH

These are restrictions for configuring the device for secure shell.

- From Cisco IOS XE Dublin 17.10.x, Key Exchange and MAC algorithms like diffie-hellman-group14-sha1, hmac-sha1, hmac-sha2-256, and hmac-sha2-512 are not supported by default and it may impact some SSH clients that only support these algorithms. However, you can add them manually if required. For information on manually adding these algorithms, see the https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/sec-vpn/b-security-vpn/m_sec-secure-shell-algorithm-ccc.html.
- 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.
- To authenticate clients with FreeRADIUS over RADSEC, you should generate an RSA key longer than 1024 bit. Use the **crypto key generate rsa general-keys exportable label** *label-name* command to achieve this.

Set up the device to run SSH

Enable SSH on the device for secure remote access.

Follow the procedure given below to set up your device to run SSH:

Before you begin

Configure user authentication for local or remote access.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the hostname and IP domain name for your device.

Example:

```
Device(config)# hostname your_hostname
```

Note
Follow this procedure only if you are configuring the device as an SSH server.

Step 3 Configure a host domain for your device.

Example:

```
Device(config)# ip domain name your_domain
```

Step 4 Enable the SSH server and generate an RSA key pair.

Example:

```
Device(config)# crypto key generate rsa
```

Generating an RSA key pair for the device automatically enables SSH.

We recommend that a minimum modulus size of 1024 bits.

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.

Note
Follow this procedure only if you are configuring the device as an SSH server.

Step 5 Exit configuration mode.

Example:

```
Device(config)# end
```

The device is now set up to run SSH, allowing secure remote access.

Configure the SSH server

Set up the SSH server to allow secure remote access to the device.

Follow the procedure given below to configure the SSH server:



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

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 (Optional) Configure the device to run SSH Version 2.

Example:

```
Device(config)# ip 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 3 Configure the SSH control parameters.

Example:

```
Device(config)# ip ssh timeout
seconds
authentication-retries
number
```

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.

By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4).

Step 4 (Optional) Use one or both of the following:

- `line vty` *line_number* [*ending_line_number*]
- **transport input ssh**

Example:

```
Device(config)# line vty
line-number
ending-line-number
```

or

```
Device(config-line)# transport input ssh
```

Enters line configuration mode to configure the virtual terminal line settings. For *line-number* and *ending-line-number* , specify a pair of lines. The range is 0 to 15.

Specifies that the device prevent non-SSH Telnet connections. This limits the router to only SSH connections.

Note
If the Virtual Terminal (VTY) lines are exhausted, Telnet or SSH will fail. You can either disconnect the Telnet or SSH sessions to free up the VTY lines, or follow the recovery steps given below to clear VTY lines and reload Telnet or SSH:

```
Device# configure terminal
Device(config)# clear line
line number
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config-line)# end
```

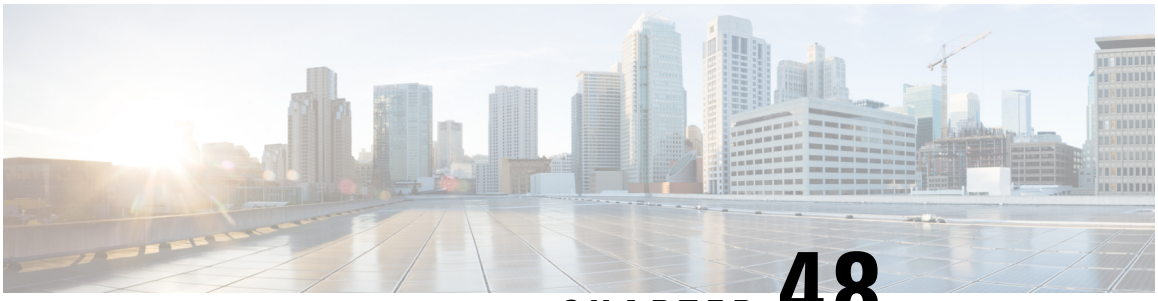
Returns to privileged EXEC mode.

The SSH server is now configured and ready for secure remote access.

Monitor the SSH configuration and status

This table displays the SSH server configuration and status.

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



CHAPTER 48

Private Shared Key

- Private preshared keys, on page 471

Private preshared keys

A private preshared key (PSK) is a security protocol for network authentication that

- enables devices to securely connect to wireless local area networks (WLANs),
- assigns unique keys to individual devices or user groups, reducing the risk of unauthorized access, and
- ensures compatibility with systems where advanced authentication methods like 802.1X are unsupported.

Understanding identity PSKs

Identity PSKs are unique PSKs created for individuals or groups of users on the same SSID. They provide the same simplicity of PSK while enabling features such as:

- Stronger security for IoT by supporting devices that do not use 802.1X.
- Easy revocation of access for a specific device or individual without affecting others.
- Scalable management of thousands of keys distributed through the AAA server.

These features make Identiy PSKs ideal for IoT, BYOD, and guest deployments.

Limitations

SSID Preshared Key Special Characters

- Special characters, such as '<' and '>' are not supported in SSID Preshared key.

SSID Preshared Key Whitespace Support

PSK supports whitespace in passwords (before or after or in-between) within double quotes only; single quotes for whitespaces are not supported.

How identity PSK authentication works

Summary

The key components involved in the process are:

- Client device: Sends an association request to connect to the access point.
- AP: Broadcasts the SSID and forwards association requests to the controller.
- Embedded Wireless Controller: Constructs RADIUS requests and relays them to the authentication server; manages computation of PSKs and applies received configuration parameters.
- RADIUS (AAA) server: Authenticates client MAC addresses, authorizes access, and returns passphrases or additional parameters as needed.

Workflow

Here are the stages you will follow for Identity PSK authentication:

1. The client device attempts to connect to the wireless network by sending an association request to the SSID broadcast by the access point.
2. The access point forwards the association request to the Embedded Wireless Controller.
3. The controller creates a RADIUS authentication request packet containing the client's MAC address and relays it to the RADIUS (AAA) server.
4. The RADIUS server performs authentication and determines whether the client is authorized.
    - If authorized, the server sends an ACCESS-ACCEPT response; otherwise, it sends an ACCESS-REJECT.
5. If the client is authorized, the RADIUS server includes the AV-pair passphrase (the identity PSK) in the response, as well as optional parameters such as username, VLAN, and Quality of Service (QoS).
6. Upon receiving the response, the controller uses the passphrase to compute the Pairwise Master Key (PMK) and apply any additional attributes for the client's session.
7. If you have multiple devices, the same passphrase can apply to all of them.
8. If a configured PSK is shorter than 15 characters (when Federal Information Processing Standard, FIPS, is enabled), the controller allows the WLAN configuration but displays the following console warning:



Note AP is allowed to join but corresponding WLAN will not be pushed to the access point.

Result

Your device will either be granted access to the network with its assigned unique PSK and relevant settings, or access will be denied based on the authentication outcome. This information could be better presented in a separate paragraph or subsection to distinctively address the outcome of the process.

Configure a PSK in a WLAN

Set up security for a pre-shared key (PSK) in a WLAN environment using command-line instructions.

Before you begin

- Configure security for a pre-shared key (PSK) in a WLAN.
- If there is no override from the AAA server, the system uses the WLAN value for authentication.
- In Federal Information Processing Standard (FIPS) and common criteria mode, ensure that the PSK WLAN uses at least 15 ASCII characters.

Procedure

Step 1 Configure terminal

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the WLAN and the SSID

Example:

```
Device(config)# wlan test-profile 4 abc
```

Configures the WLAN and SSID.

Step 3 Disable security AKM for 802.11

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Disables security AKM for 802.11x.

Step 4 Configure the security type PSK

Example:

```
Device(config-wlan)# security wpa akm psk
```

Configures the security type PSK.

Step 5 Configure PSK authenticated key management (AKM) shared key

Example:

```
Device(config-wlan)# security wpa akm psk set-key asci 0
```

Configures the PSK authenticated key management (AKM) shared key.

Note
You must set the **psk set-key** before configuring AKM PSK.

Step 6 Configure PSK support

Example:

```
Device(config-wlan)# security wpa akm psk
```

Configures PSK support.

Step 7 Specify MAC filtering in WLAN

Example:

```
Device(config-wlan)# mac-filtering test1
```

Specifies MAC filtering in a WLAN.

The WLAN is configured to use a pre-shared key for client authentication.

Configure a PSK in a WLAN using GUI

Configure a pre-shared key (PSK) for a WLAN.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **Wireless Networks** page, click **Security** tab.

Step 3 In the **Layer 2** window that is displayed, go to the **WPA Parameters** section.

Step 4 From the **Auth Key Mgmt** drop-down, select the PSK format and type.

Step 5 Enter the Pre-Shared Key in hexadecimal characters.

- If you selected the PSK format as HEX, the key length must be exactly 64 characters.
- If you selected the PSK format as ASCII, ensure the key length ranges from 8 to 63 characters.

After configuring the key, these details are not visible due to security reasons. Even if you click on the eye icon next to the Pre-Shared Key box, the details remain hidden.

Step 6 Click **Save & Apply to Device**.

The PSK is configured and applied to the selected WLAN.

What to do next

Check the WLAN connectivity to ensure successful configuration.

Apply a policy profile to a WLAN (GUI)

This task helps you configure WLAN by applying policy profiles for better network management.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the **Manage Tags** page, click **Policy** tab.

Step 3 Click **Add** to view the **Add Policy Tag** window.

Step 4 Enter the policy tag name and its description.

Step 5 Click **Add** to map WLAN and policy.

Step 6 Choose the WLAN profile to map with the appropriate policy profile, then click the tick icon.

Step 7 Click **Save & Apply to Device**.

Once the task is completed, you successfully map the WLAN to the chosen policy profile

What to do next

Verify the changes by checking the applied profiles under Wireless Local Area Network (WLAN) settings.

Monitor network performance to confirm the policy effectiveness.

Apply a policy profile to a WLAN using CLI

The goal of this task is to effectively apply a policy profile to a WLAN using CLI commands, which helps manage network configurations efficiently.

Procedure

Step 1 Enter global configuration mode

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Configure the default policy profile

Example:

```
Device(config)# wireless profile policy policy-iot
```

Configures the default policy profile.

Step 3 Configure AAA override to apply policies coming from the AAA server

Example:

```
Device(config-wireless-policy)# aaa-override
```

Configures AAA override to apply policies coming from the AAA server or ISE, the Cisco Identity Services Engine server.

Once you apply the policy profile, configure the WLAN with the specified policy settings to enable the expected network management and security features.

What to do next

You should verify the applied configuration by checking the WLAN settings to ensure all policies are correctly enforced.

- Confirm connectivity and performance with the new settings.

Verify a private PSK

To verify the configuration of a WLAN and a client, use the following **show** commands

Verify wireless client configuration

```
Device# show wlan id 2


WLAN Profile Name     : test_ppsk
================================================
Identifier                                       : 2
Network Name (SSID)                              : test_ppsk
Status                                           : Enabled
Broadcast SSID                                   : Enabled
Universal AP Admin                               : Disabled
Max Associated Clients per WLAN                  : 0
Max Associated Clients per AP per WLAN           : 0
Max Associated Clients per AP Radio per WLAN     : 0
Number of Active Clients                         : 0
Exclusionlist Timeout                            : 60
CHD per WLAN                                     : Enabled
Interface                                        : default
Multicast Interface                              : Unconfigured
WMM                                              : Allowed
WifiDirect                                       : Invalid
Channel Scan Defer Priority:
  Priority (default)                             : 4
  Priority (default)                             : 5
  Priority (default)                             : 6
Scan Defer Time (msecs)                          : 100
Media Stream Multicast-direct                    : Disabled
CCX - AironetIe Support                          : Enabled
CCX - Diagnostics Channel Capability             : Disabled
Peer-to-Peer Blocking Action                     : Disabled
Radio Policy                                     : All
DTIM period for 802.11a radio                    : 1
DTIM period for 802.11b radio                    : 1
Local EAP Authentication                         : Disabled
Mac Filter Authorization list name               : test1
Accounting list name                             : Disabled
802.1x authentication list name                  : Disabled
Security
    802.11 Authentication                        : Open System
    Static WEP Keys                              : Disabled
    802.1X                                       : Disabled
    Wi-Fi Protected Access (WPA/WPA2)            : Enabled
        WPA (SSN IE)                             : Disabled
        WPA2 (RSN IE)                            : Enabled
            TKIP Cipher                          : Disabled
```

```
            AES Cipher                          : Enabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Enabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            PMF dot1x                           : Disabled
            PMF PSK                             : Disabled
    CCKM TSF Tolerance                          : 1000
    FT Support                                  : Disabled
        FT Reassociation Timeout                : 20
        FT Over-The-DS mode                     : Enabled
    PMF Support                                 : Disabled
        PMF Association Comeback Timeout        : 1
        PMF SA Query Time                       : 200
    Web Based Authentication                    : Disabled
    Conditional Web Redirect                    : Disabled
    Splash-Page Web Redirect                    : Disabled
    Webauth On-mac-filter Failure               : Disabled
    Webauth Authentication List Name            : Disabled
    Webauth Parameter Map                       : Disabled
    Tkip MIC Countermeasure Hold-down Timer     : 60
Call Snooping                                   : Disabled
Passive Client                                  : Disabled
Non Cisco WGB                                   : Disabled
Band Select                                     : Disabled
Load Balancing                                  : Disabled
Multicast Buffer                                : Disabled
Multicast Buffer Size                           : 0
IP Source Guard                                 : Disabled
Assisted-Roaming
    Neighbor List                               : Disabled
    Prediction List                             : Disabled
    Dual Band Support                           : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                  : Disabled
    BSS Max Idle                                : Disabled
        Protected Mode                          : Disabled
    Traffic Filtering Service                   : Disabled
    BSS Transition                              : Enabled
        Disassociation Imminent                 : Disabled
            Optimised Roaming Timer             : 40
            Timer                               : 200
    WNM Sleep Mode                              : Disabled
802.11ac MU-MIMO                                : Disabled
```

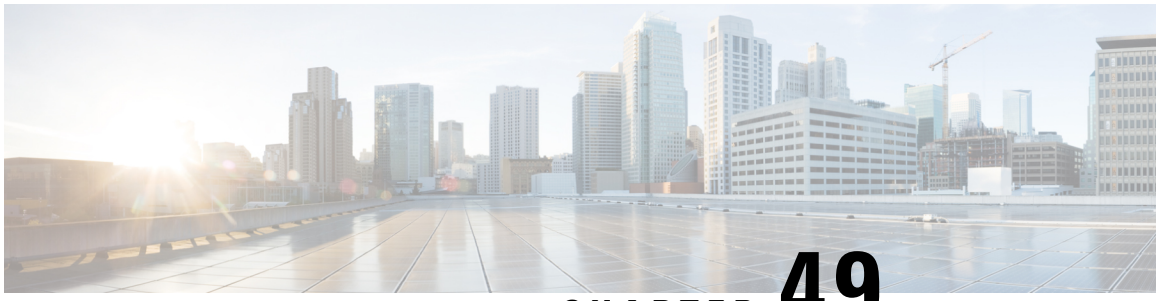
Commands for PSK verification

```
Device# show wireless client mac-address a886.adb2.05f9 detail


Client MAC Address : a886.adb2.05f9
Client IPv4 Address : 9.9.58.246
Client Username : A8-86-AD-B2-05-F9
AP MAC Address : c025.5c55.e400
AP Name: saurabh-3600
AP slot : 1
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : default-flex-profile
Wireless LAN Id : 6
Wireless LAN Name: SSS_PPSK
```

```
BSSID : c025.5c55.e40f
Connected For : 280 seconds
Protocol : 802.11n - 5 GHz
Channel : 60
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Session Timeout : 320 sec (Remaining time: 40 sec)
Input Policy Name  :
Input Policy State : None
Input Policy Source : None
Output Policy Name  :
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Current Rate : m22
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 09/27/2017 16:32:25 IST
Policy Manager State: Run
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 280 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : PSK
AAA override passphrase: Yes
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : Not Applicable
VLAN : 58
Access VLAN : 58
Anchor VLAN : 0
WFD capable : No
Manged WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Interface        : capwap_90000005
  IIF ID           : 0x90000005
  Device Type      : Apple-Device
  Protocol Map     : 0x000001
  Authorized       : TRUE
  Session timeout  : 320
  Common Session ID: 1F38090900000005DC30088EA
  Acct Session ID  : 0x00000000
  Auth Method Status List
        Method : MAB
                SM State        : TERMINATE
                Authen Status   : Success
  Local Policies:
        Service Template : wlan_svc_default-policy-profile (priority 254)
                Absolute-Timer   : 320
                VLAN             : 58
  Server Policies:
```

```
  Resultant Policies:
               VLAN             : 58
               Absolute-Timer   : 320
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Local
FlexConnect Dhcp Status : Local
FlexConnect Authentication : Central
FlexConnect Central Association : No
Client Statistics:
  Number of Bytes Received : 59795
  Number of Bytes Sent : 21404
  Number of Packets Received : 518
  Number of Packets Sent : 274
  Number of EAP Id Request Msg Timeouts :
  Number of EAP Request Msg Timeouts :
  Number of EAP Key Msg Timeouts :
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -32 dBm
  Signal to Noise Ratio : 58 dB
Fabric status : Disabled
```



CHAPTER 49

Multi-Preshared Key

- Multi-preshared key, on page 481
- Procedures, on page 483

Multi-preshared key

A multi-preshared key (multi-PSK) is a wireless security feature that

- Allows multiple pre-shared keys (PSKs) to be configured for a single SSID
- Enables any configured PSK to grant access to the same wireless network
- Improves network flexibility by supporting concurrent user groups or devices with different credentials.

PSK: A pre-shared key is a password or passphrase used to authenticate clients on a wireless network.

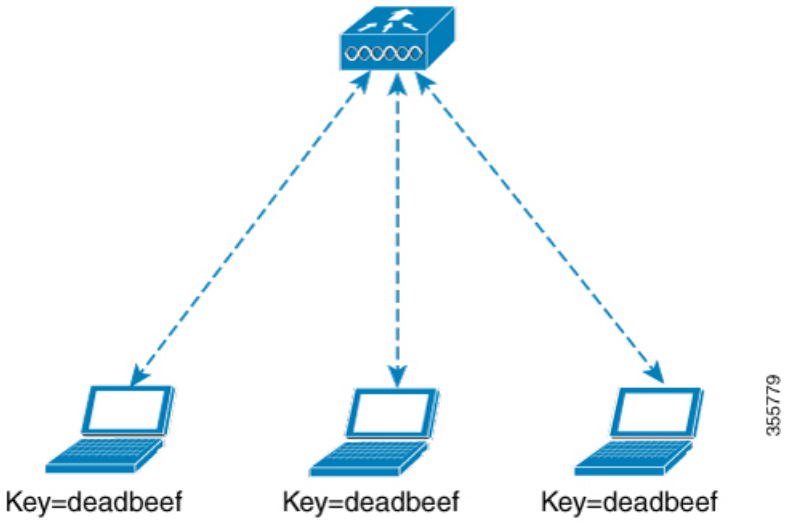
Supporting analogy: keycards to the same door

Having multi-PSk keys for a single SSID is like giving different colored keycards to various teams. Each keycard opens the same door, but you can manage which card goes to which team. This approach increases flexibility and security compared to using one generic key.

Comparing traditional PSK with

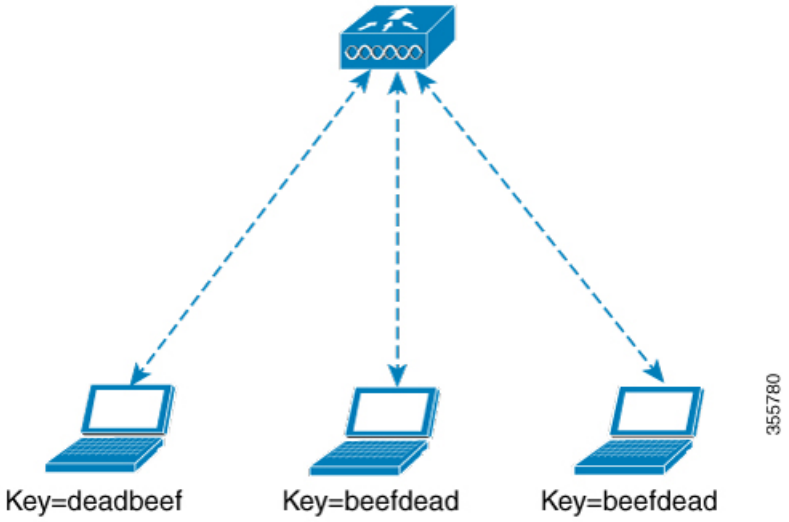
In a traditional PSK, all the clients joining the network use the same password as shown in the figure.

Figure 8: Traditional PSK



But with multi-PSK, client can use any of the configured pre-shared keys to connect to the network as shown in the figure.

Figure 9: Multi-PSK



In the Multi-PSK example, two passwords—deadbeef and beefdead—are configured for the same SSIDIn this scenario, clients can connect to the network using either of the passwords.

Note

-
- Multi-PSK is different from iPSK. In iPSK, the PSK password comes from ISE authorization policy, so MAB is required. Multi-PSK uses a pool of passwords locally configured in WLAN, so Identity Service Engine is not used.

Feature	Traditional PSK	Multi-PSK	Identity PSK (iPSK)
Number of PSKs per SSID	One shared key	Multiple keys (up to five)	Unique key per user or per group
Use case flexibility	All users share one credential	Separate keys for groups or devices	Per-user or per-group credentials for high granularity
Example	All staff share same key	Staff, guests, IoT devices have different keys	Each staff member, contractor, or device gets its own key
Key management	Single change affects all users	Changes can target specific groups or devices	Changes can target specific users or devices; policy-driven from ISE
Security granularity	Lowest — one compromise affects all	Better — compromise isolated to that PSK group	Highest — compromise isolated to individual user or device and fully policy-based

Restrictions

- In central authentication flex mode, the standalone AP allows client join with the highest priority PSK (priority 0 key). New clients that do not use the highest priority PSK are rejected during the standalone mode.
- Multi-PSK does not support local authentication.

Procedures

Configure a multi-preshared key (GUI)

Configure a WLAN to use a multi-PSK through the controller's GUI.

Before you begin

Know the required security settings such as Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), PSK, and so on for your WLAN.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **Wireless Networks** page, click the name of the WLAN.

Step 3 In the **Edit WLAN** window, click the **Security** tab.

Step 4 In the **Layer2** tab, choose the **Layer2 Security Mode** from the options:

- None: No Layer 2 security
- 802.1X: WEP 802.1X data encryption type
- WPA + WPA2: Wi-Fi Protected Access
- Static WEP: Static WEP encryption parameters
- Static WEP+802.1X: Both Static WEP and 802.1X parameters

Parameters	Description
802.1X	
WEP Key Size	Choose the key size. The available values are: • None • 40 bits • 104 bits
WPA + WPA2	
Protected Management Frame	Possible values are: • Disabled • Optional • Required
WPA Policy	Check the check box to enable WPA policy.
WPA Encryption	Choose the WPA encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.
WPA2 Policy	Check the check box to enable WPA2 policy.
WPA2 Encryption	Choose the WPA2 encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.

Parameters	Description
Auth Key Mgmt	Possible rekeying mechanism options: • 802.1X • FT + 802.1X • PSK: You must specify the PSK format and a preshared key • Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • FT + 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value
Static WEP	
Key Size	Possible key size options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
Static WEP + 802.1X	
Key Size	Possible key size options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.

Parameters	Description
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
WEP Key Size	Choose from the WEP key sizes: • None • 40 bits • 104 bits

Step 5 Click **Save & Apply to Device**.

The WLAN is updated with the selected multi-PSK security settings.

Configure a multi-preshared key (CLI)

Configure a WLAN to use a multi-PSK through the controller's CLI.

Before you begin

Know the required security settings (WEP, WPA, PSK, and so on.) for your WLAN.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure WLAN and SSID.

Example:

```
Device(config)# wlan mywlan 1 SSID_name
```

Step 3 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm
                      dot1x
```

Step 4 Configure PSK.

Example:

```
Device(config-wlan)# security wpa akm psk
```

Step 5 Configure multi-PSK.

Example:

```
Device(config-wlan)# security wpa wpa2 mpsk
```

Step 6 Configure PSK priority and all its related passwords using the **priority** *priority_value* **set-key** {**ascii** [**0** | **8**] *pre-shared-key* | **hex** [**0** | **8**] *pre-shared-key*} command.

Example:

```
Device(config-mpsk)# priority 0 set-key ascii 0 deadbeef
```

The *priority_value* ranges from 0 to 4.

Note
You need to configure **priority 0** key for multi-PSK.

Step 7 Enable WLAN.

Example:

```
Device(config-mpsk)# no shutdown
```

Step 8 Exit WLAN configuration mode and returns to configuration mode.

Example:

```
Device(config-wlan)# exit
```

Step 9 Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example:

```
Device(config)# end
```

The WLAN is updated with the selected multi-PSK security settings.

Verify multi-PSK configurations

To verify the configuration of a WLAN and a client, use the following command:

```
Device# show wlan id 8
WLAN Profile Name     : wlan_8
================================================
Identifier                                    : 8
Network Name (SSID)                           : ssid_8
Status                                        : Enabled
Broadcast SSID                                : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
Number of Active Clients                      : 0
CHD per WLAN                                  : Enabled
Multicast Interface                           : Unconfigured
WMM                                           : Allowed
```

```
WifiDirect                                      : Invalid
Channel Scan Defer Priority:
  Priority (default)                            : 5
  Priority (default)                            : 6
Scan Defer Time (msecs)                         : 100
Media Stream Multicast-direct                   : Disabled
CCX - AironetIe Support                         : Enabled
CCX - Diagnostics Channel Capability            : Disabled
Peer-to-Peer Blocking Action                    : Disabled
Radio Policy                                    : All
DTIM period for 802.11a radio                   : 1
DTIM period for 802.11b radio                   : 1
Local EAP Authentication                        : Disabled
Mac Filter Authorization list name              : Disabled
Mac Filter Override Authorization list name     : Disabled
Accounting list name                            :
802.1x authentication list name                 : Disabled
802.1x authorization list name                  : Disabled
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    802.1X                                      : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)      : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Enabled
            MPSK                                : Enabled
            AES Cipher                          : Enabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP256 Cipher                      : Disabled
        WPA3 (WPA3 IE)                          : Disabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Enabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            FT SAE                              : Disabled
            PMF dot1x                           : Disabled
            PMF PSK                             : Disabled
            SAE                                 : Disabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    CCKM TSF Tolerance                          : 1000
    FT Support                                  : Adaptive
        FT Reassociation Timeout                : 20
        FT Over-The-DS mode                     : Enabled
    PMF Support                                 : Disabled
        PMF Association Comeback Timeout        : 1
        PMF SA Query Time                       : 200
    Web Based Authentication                    : Disabled
    Conditional Web Redirect                    : Disabled
    Splash-Page Web Redirect                    : Disabled
    Webauth On-mac-filter Failure               : Disabled
    Webauth Authentication List Name            : Disabled
    Webauth Authorization List Name             : Disabled
    Webauth Parameter Map                       : Disabled
    Tkip MIC Countermeasure Hold-down Timer     : 60
Non Cisco WGB                                   : Disabled
Band Select                                     : Enabled
Load Balancing                                  : Disabled
Multicast Buffer                                : Disabled
Multicast Buffer Size                           : 0
```

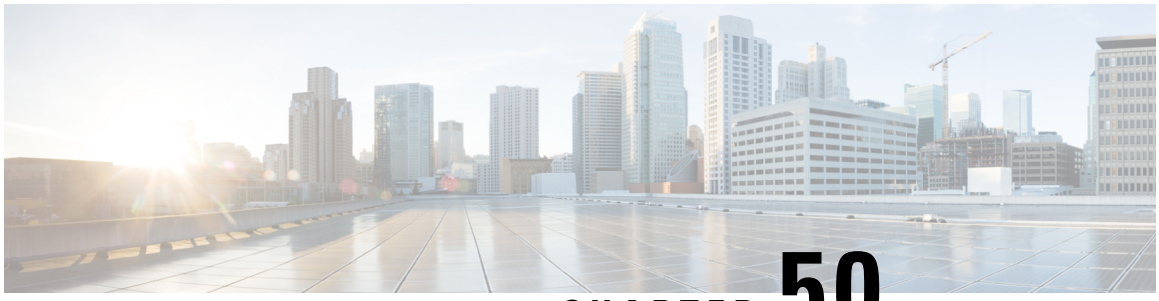
```
IP Source Guard                                 : Disabled
Assisted-Roaming
    Neighbor List                               : Disabled
    Prediction List                             : Disabled
    Dual Band Support                           : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                  : Disabled
    BSS Max Idle                                : Disabled
        Protected Mode                          : Disabled
    Traffic Filtering Service                   : Disabled
    BSS Transition                              : Enabled
        Disassociation Imminent                 : Disabled
            Optimised Roaming Timer             : 40
            Timer                               : 200
    WNM Sleep Mode                              : Disabled
802.11ac MU-MIMO                                : Disabled
802.11ax paramters
    OFDMA Downlink                              : unknown
    OFDMA Uplink                                : unknown
    MU-MIMO Downlink                            : unknown
    MU-MIMO Uplink                              : unknown
    BSS Color                                   : unknown
    Partial BSS Color                           : unknown
    BSS Color Code                              :
```

To view the WLAN details, use the following command:

```
Device# show run wlan
wlan wlan_8 8 ssid_8
 security wpa psk set-key ascii 0 deadbeef
 no security wpa akm dot1x
 security wpa akm psk
 security wpa wpa2 mpsk
  priority 0 set-key ascii 0 deadbeef
  priority 1 set-key ascii 0 deaddead
  priority 2 set-key ascii 0 d123d123
  priority 3 set-key hex 0 023456789012345678901234567890123456789012345678901234567890123456789012345678901234

  priority 4 set-key hex 0 123456789012345678901234567890123456789012345678901234567890123456789012345678901234

 no shutdown
```



CHAPTER 50

Multiple Authentications for a Client

- Multiple authentications for a client, on page 491
- Configure multiple authentications for a client, on page 492
- Verify multiple authentication configurations, on page 499

Multiple authentications for a client

Multiple authentications for a client is a network security feature that

- enables both Layer 2 (L2) and Layer 3 (L3) authentication for wireless client devices
- enhances security by requiring clients to complete multiple types of authentication before connecting, and
- applies only to regular client devices joining a wireless network.

Additional information

- You can enable both L2 and L3 authentication for a given SSID.
- Multiple authentications feature is not supported for guest or specialized client types.

Supported combination of authentications for a client

The Multiple Authentications for a Client feature supports multiple combination of authentications for a given client configured in the WLAN profile.

The Multiple Authentications for a Client feature supports multiple combinations of authentications for clients configured in the WLAN profile.

The table outlines the supported combinations of authentications for clients.

Layer 2	Layer 3	Supported
MAB	CWA	Yes
MAB Failure	LWA	Yes
802.1X	CWA	Yes

PSK	CWA	Yes
iPSK + MAB	CWA	Yes
iPSK	LWA	No
MAB Failure + PSK	LWA	No
MAB Failure + PSK	CWA	No

Unsupported combinations

The table outlines the combination of authentications on MAC failure that are not supported on a given client:

Authentication Types	Foreign	Anchor	Supported
WPA3-OWE+LWA	Cisco AireOS	Cisco Catalyst 9800 Controller	No
WPA3-SAE+LWA	Cisco AireOS	Cisco Catalyst 9800 Controller	No

Jumbo frame support for RADIUS packets

RADIUS packets are fragmented according to the MTU of the egress interface when you meet all of these conditions.

- The command **ip radius source-interface** is configured under the relevant AAA group server radius group to point to the egress interface.
- The **ip mtu NNN** command is configured on the egress interface.

Note If you set the MTU of the source interface to less than 1500 bytes, additional fragmentation can occur. This may cause packet drops by upstream devices, such as firewalls and load balancers. Authentication failures may result. Verify these configurations during upgrades to prevent these issues.

\

Configure multiple authentications for a client

Configure a WLAN for 802.1X and LWA

Set up a WLAN to enforce strong user authentication with 802.1X and provide local web authentication using the controller interface.

Perform this task when you want to secure WLAN user access with 802.1X credentials and enable local web authentication for guest or additional user workflows.

Before you begin

Confirm that WLAN profiles and the necessary authentication lists are already created or available.

Procedure

Step 1	Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2	Select the required WLAN from the list of WLANs displayed.
Step 3	Choose **Security** > **Layer2** tab.
Step 4	Select the security method from the **Layer 2 Security Mode** drop-down list.
Step 5	In the **Auth Key Mgmt**, check the **802.1x** check box.
Step 6	Check the **MAC Filtering** check box to enable the feature.
Step 7	After MAC Filtering is enabled, from the **Authorization List** drop-down list, choose an option.
Step 8	Choose **Security** > **Layer3** tab.
Step 9	Check the **Web Policy** check box to enable the web authentication policy.
Step 10	From the **Web Auth Parameter Map** and the **Authentication List** drop-down lists, choose an option.
Step 11	Click **Update & Apply to Device**.

The WLAN is now configured to use both 802.1X authentication and local web authentication policies.

Configure a WLAN for 802.1X and LWA (CLI)

Set up a WLAN to enforce strong user authentication with 802.1X and provide local web authentication using the controller interface.

Perform this task when you want to secure WLAN user access with 802.1X credentials and enable local web authentication for guest or additional user workflows.

Before you begin

Confirm that WLAN profiles and the necessary authentication lists are already created or available.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN.

Example:

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.

- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

Step 3 Enable security authentication list for dot1x security.

Example:

```
Device(config-wlan)# security dot1x authentication-list auth-list-name
```

The configuration is similar for all dot1x security WLANs.

Step 4 Enable web authentication.

Example:

```
Device(config-wlan)# security web-auth
```

Step 5 Enable authentication list for dot1x security.

Example:

```
Device(config-wlan)# security web-auth authentication-list default authenticate-list-name
```

Step 6 Map the parameter map name.

Example:

```
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

If a parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

Step 7 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

The WLAN is now configured to use both 802.1X authentication and local web authentication policies.

Configure WLAN for Preshared Key (PSK) and LWA (GUI)

Set up a WLAN to use both Preshared Key (PSK) and local web authentication via the graphical user interface.

Use this procedure when you want to enforce both PSK and local web authentication for user access to your wireless network.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Select the required WLAN.

Step 3 Choose **Security** > **Layer2** tab.

Step 4 Select the security method from the **Layer 2 Security Mode** drop-down list.

Step 5 In the Auth Key Mgmt, uncheck the **802.1x** check box.

Step 6 Check the **PSK** check box.

Step 7 Enter the **Pre-Shared Key**. Next, choose the PSK Format from the **PSK Format** drop-down list. Then, choose the PSK Type from the **PSK Type** drop-down list.

Step 8 Choose **Security** > **Layer3** tab.

Step 9 Check the **Web Policy** checkbox to enable web authentication policy.

Step 10 Choose the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list. Then, choose the authentication list from the **Authentication List** drop-down list.

Step 11 Click **Update & Apply to Device**.

The selected WLAN now enforces both PSK and local web authentication settings

Configure WLAN for Preshared Key (PSK) and LWA (CLI)

Set up a WLAN to use both Preshared Key (PSK) and local web authentication.

Use this procedure when you want to enforce both PSK and local web authentication for user access to your wireless network.

Procedure

	Command or Action	Purpose
Step 1	Enter global configuration mode. **Example:** `Device# configure terminal`	
Step 2	Configure the WLAN. **Example:** `Device(config)# wlan wlan-name wland-id SSID-name`	• **wlan-name** is the name of the configured WLAN. • **wlan-id** is the WLAN identifier. The range is one to 512. • **SSID-name** is the SSID name which can have up to 32 alphanumeric characters. If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.
Step 3	Configure the PSK shared key using the **security wpa psk set-key** *ascii/hex key password* command. **Example:** `Device(config-wlan)# security wpa psk set-key ascii 0 PASSWORD`	
Step 4	Disable security AKM for dot1x. **Example:**	

	Command or Action	Purpose
	`Device(config-wlan)# no security wpa akm dot1x`	
Step 5	Configure the PSK support. **Example:** `Device(config-wlan)# security wpa akm psk`	
Step 6	Enable web authentication for WLAN. **Example:** `Device(config-wlan)# security web-auth`	
Step 7	Enable authentication list for dot1x security. **Example:** `Device(config-wlan)# security web-auth authentication-list authenticate-list-name`	
Step 8	Configure the parameter map. **Example:** `(config-wlan)# security web-auth parameter-map parameter-map-name`	If parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

The selected WLAN now enforces both PSK and local web authentication settings.

Configure WLAN for PSK or iPSK and CWA

Configure WLAN for PSK or iPSK and CWA (GUI)

Configure a WLAN to use either a Pre-Shared Key (PSK) or Identity Pre-Shared Key (iPSK) for authentication, and enable central web authentication for enhanced wireless security.

Use this task to set up secure WLAN access for users and devices, combining key-based authentication with web-based login, using your wireless controller's GUI.

Before you begin

Gather required PSK or iPSK values and authentication lists, as needed.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Select the required WLAN.

Step 3 Choose **Security** > **Layer2** tab.

Step 4 Select the security method from the **Layer 2 Security Mode** drop-down list.

Step 5 In the **Auth Key Mgmt**, uncheck the **802.1x** check box.

Step 6 Check the **PSK** check box.

Step 7 Enter the **Pre-Shared Key**. Then, choose the PSK Format from the **PSK Format** drop-down list and the PSK Type from the **PSK Type** drop-down list.

Step 8 Check the **MAC Filtering** check box to enable the feature.

Step 9 With MAC Filtering enabled, choose the Authorization List from the **Authorization List** drop-down list.

Step 10 Choose **Security** > **Layer3** tab.

Step 11 Check the **Web Policy** checkbox to enable web authentication policy.

Step 12 Choose the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list and the authentication list from the **Authentication List** drop-down list.

Step 13 Click **Update &Apply to Device**.

The WLAN is now updated to use PSK or iPSK authentication and central web authentication, combining key-based access with web-based login for network users.

Configure WLAN for PSK or iPSK and CWA (CLI)

Configure a WLAN to use either a Pre-Shared Key (PSK) or Identity Pre-Shared Key (iPSK) for authentication, and enable central web authentication for enhanced wireless security.

Use this task to set up secure WLAN access for users and devices, combining key-based authentication with web-based login, using your wireless controller's CLI.

Before you begin

Gather required PSK or iPSK values and authentication lists, as needed.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN.

Example:

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

Step 3 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Step 4 Configure the PSK AKM shared key using the **security wpa psk set-key** *ascii/hex key password* command.

Example:

```
Device(config-wlan)# security wpa psk set-key ascii 0 PASSWORD
```

Step 5 Set the MAC filtering parameters.

Example:

```
Device(config-wlan)# mac-filtering auth-list-name
```

The WLAN is now updated to use PSK or iPSK authentication and central web authentication, combining key-based access with web-based login for network users.

Apply a policy profile to a WLAN

Configure and activate a policy profile for a WLAN using CLI.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the default policy profile.

Example:

```
Device(config)# wireless profile policy policy-iot policy-profile-name
```

Step 3 Configure AAA override to apply policies coming from the AAA or ISE servers.

Example:

```
Device(config-wireless-policy)# aaa-override
```

Step 4 Configure NAC in the policy profile.

Example:

```
Device(config-wireless-policy)# nac
```

Step 5 Shutdown the WLAN.

Example:

```
Device(config-wireless-policy)# no shutdown
```

Step 6 Return to privileged EXEC mode.

Example:

```
Device(config-wireless-policy)# end
```

The policy profile is applied to the WLAN and operational.

Verify multiple authentication configurations

Layer 2 authentication

After L2 authentication (Dot1x) is complete, the client is moved to state.

To verify the client state after L2 authentication, use these commands:

```
Device# show wireless client summary
Number of Local Clients: 1
MAC Address  AP Name  WLAN  State  Protocol  Method  Role
-------------------------------------------------------------------------------------------------------------------

58ef.68b6.aa60 ewlc1_ap_1   3  Webauth Pending    11n(5)        Dot1x  Local
Number of Excluded Clients: 0
```

```
Device# show wireless client mac-address <mac_address> detail

Auth Method Status List

Method: Dot1x
Webauth State: Init
Webauth Method: Webauth
Local Policies:
Service Template: IP-Adm-V6-Int-ACL-global (priority 100)
URL Redirect ACL: IP-Adm-V6-Int-ACL-global
Service Template: IP-Adm-V4-Int-ACL-global (priority 100)
URL Redirect ACL: IP-Adm-V4-Int-ACL-global
Service Template: wlan_svc_default-policy-profile_local (priority 254)
Absolute-Timer: 1800
VLAN: 50
```

```
Device# show platform software wireless-client chassis active R0

        ID  MAC Address      WLAN  Client     State
-----------------------------------------------------------------------------------
  0xa0000003      58ef.68b6.aa60    3             L3          Authentication
```

```
Device# show platform software wireless-client chassis active F0

    ID     MAC Address   WLAN  Client     State  AOM ID    Status
-------------------------------------------------------------------------------------------------
0xa0000003    58ef.68b6.aa60    3           L3          Authentication.         730.
Done
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
summary

Client Type Abbreviations:
RG - REGULAR   BLE - BLE
HL - HALO    LI - LWFL INT

Auth State Abbrevations:
UK - UNKNOWN    IP - LEARN    IP IV - INVALID
L3 - L3 AUTH RN - RUN

Mobility State Abbreviations:
UK - UNKNOWN       IN - INIT
LC - LOCAL                AN - ANCHOR
FR - FOREIGN          MT - MTE
IV - INVALID
```

```
EoGRE Abbreviations:
N – NON EOGRE Y - EOGRE

CPP IF_H   DP IDX       MAC Address      VLAN   CT   MCVL AS MS E   WLAN      POA
------------------------------------------------------------------------------------
0X49     0XA0000003    58ef.68b6.aa60     50    RG     0  L3 LC N wlan-test 0x90000003
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath summary
Vlan   DP IDX         MAC Address      VLAN   CT   MCVL AS MS E   WLAN      POA
------------------------------------------------------------------------------------
0X49   0xa0000003    58ef.68b6.aa60     50    RG     0  L3 LC N wlan-test 0x90000003
```

Layer 3 Authentication

Once L3 authentication is successful, the client is moved to *Run* state.

To verify the client state after L3 authentication, use these commands:

```
Device# show wireless client summary

Number of Local Clients: 1
MAC Address  AP Name  WLAN  State  Protocol  Method  Role
---------------------------------------------------------------------------------------------------------

58ef.68b6.aa60   ewlc1_ap_1  3     Run    11n(5)   Web Auth      Local
Number of Excluded Clients: 0
```

```
Device# show wireless client mac-address 58ef.68b6.aa60 detail

Auth Method Status List

Method: Web Auth
Webauth State: Authz
Webauth Method: Webauth
Local Policies:
Service Template: wlan_svc_default-policy-profile_local (priority 254)
Absolute-Timer: 1800
VLAN: 50

Server Policies:

Resultant Policies:
VLAN: 50
Absolute-Timer: 1800
```

```
Device# show platform software wireless-client chassis active R0

ID          MAC Address     WLAN    Client State
-------------------------------------------------
0xa0000001 58ef.68b6.aa60    3          Run
```

```
Device# show platform software wireless-client chassis active f0

ID         MAC Address       WLAN    Client State   AOM ID.  Status
-----------------------------------------------------------------------
0xa0000001 58ef.68b6.aa60.   3          Run         11633    Done
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
summary

Client Type Abbreviations:
RG – REGULAR   BLE – BLE
HL - HALO      LI – LWFL INT

Auth State Abbrevations:
UK – UNKNOWN    IP – LEARN    IP IV – INVALID
```

```
L3 - L3 AUTH RN - RUN
Mobility State Abbreviations:
UK - UNKNOWN       IN - INIT
LC - LOCAL         AN - ANCHOR
FR - FOREIGN       MT - MTE
IV - INVALID
EoGRE Abbreviations:
N - NON EOGRE Y - EOGRE

CPP IF_H   DP IDX       MAC Address   VLAN  CT  MCVL AS MS E   WLAN     POA
------------------------------------------------------------------------------
0X49     0XA0000003    58ef.68b6.aa60  50   RG   0   RN LC N wlan-test 0x90000003

Device# show platform hardware chassis active qfp feature wireless wlclient datapath summary

Vlan   pal_if_hd1        mac           Input Uidb     Output Uidb
-----------------------------------------------------------------
50     0xa0000003    58ef.68b6.aa60     95929            95927
```

Verifying PSK+Webauth Configuration

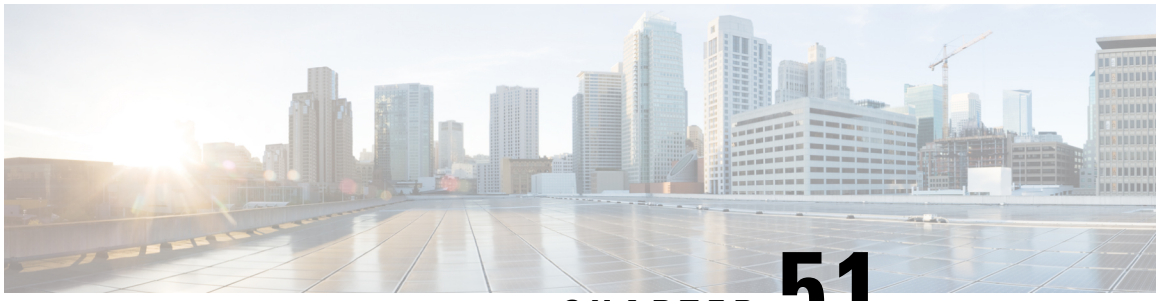
```
Device# show wlan summary

Load for five secs: 0%/0%; one minute: 0%; five minutes: 0%
Time source is NTP, 12:08:32.941 CEST Tue Oct 6 2020


Number of WLANs: 1

ID Profile Name SSID Status Security
-------------------------------------------------------------------------------------------------------------------------------

23 Gladius1-PSKWEBAUTH Gladius1-PSKWEBAUTH UP [WPA2][PSK][AES],[Web Auth]
```



CHAPTER 51

Locally Significant Certificates

- Locally significant certificates, on page 503
- Provisioning locally significant certificates, on page 506
- AP certificate renewal (LSC), on page 519
- MIC and LSC APs joining the controller, on page 526
- LSC fallback APs, on page 531
- Configuring controller self-signed certificate for wireless AP join, on page 533

Locally significant certificates

A locally significant certificate (LSC) is a PKI certificate that

- provides mutual authentication between Cisco Embedded Wireless Controller on Catalyst Access Points and Lightweight Access Points (LAPs)
- generates certificates directly on APs and embedded wireless controllers, and
- enables organizations to use their own PKI for better security control and Certificate Authority (CA) management.

LSC provisioning and communication process

You need to provision the new LSC certificate on the embedded wireless controller and then the Lightweight Access Point (LAP) from the CA Server.

The LAP communicates with the embedded wireless controller using the CAPWAP protocol. Any request to sign the certificate and issue the CA certificates for LAP and embedded wireless controller itself must be initiated from the embedded wireless controller. The LAP does not communicate directly with the CA server. The CA server details must be configured on the embedded wireless controller and must be accessible.

The embedded wireless controller makes use of the Simple Certificate Enrollment Protocol (SCEP) to forward certReqs generated on the devices to the CA and makes use of SCEP again to get the signed certificates from the CA.

The SCEP is a certificate management protocol that the PKI clients and CA servers use to support certificate enrollment and revocation. It is widely used in Cisco and supported by many CA servers. In SCEP, HTTP is used as the transport protocol for the PKI messages. The primary goal of SCEP is the secure issuance of certificates to network devices. SCEP is capable of many operations, but for our release, SCEP is utilized for the following operations:

- CA and Router Advertisement (RA) Public Key Distribution
- Certificate Enrollment

Certificate provisioning in controllers

Certificate provisioning in controllers is a security process that you use to:

- Install new Locally Significant Certificate (LSC) certificates (both Certificate Authority (CA) certificates and device certificates) on the controller
- Use Simple Certificate Enrollment Protocol (SCEP) to receive Certificate Authority (CA) certificates from the CA server during initial setup when the controller has no certificates, and
- Push the same Certificate Authority (CA) certificates to Access Points (APs) when you provision the Access Points (APs) with Locally Significant Certificates (LSCs).

RSA keypair recommendations

These recommendations apply when configuring PKI certificates with RSA keypairs:

Note We recommend that you use a new RSA keypair name for the newly configured PKI certificate. If you want to reuse an existing RSA keypair name (that is associated with an old certificate) for a new PKI certificate, do either of the following:

- Do not regenerate a new RSA keypair with an existing RSA keypair name. Instead, reuse the existing RSA keypair name. All certificates associated with the existing RSA keypair become invalid when you regenerate a new RSA keypair with an existing name.
- Remove the old PKI certificate configurations before reusing the existing RSA keypair name for the new PKI certificate.

How device certificate enrollment works

When your Lightweight Access Point (LAP) and controller request Certificate Authority (CA)-signed certificates, they send the certRequest as a PKCS#10 message. The Certificate Authority (CA) receives a PKCS#10 certRequest. The Certificate Authority (CA) requires additional information to authenticate the identity of the requester and verify that the request is unaltered.

Summary

The key components involved in the device certificate enrollment process are:

- Lightweight Access Point (LAP) and controller: request Certificate Authority (CA)-signed certificates by sending PKCS#10 messages. These messages contain Subject Name, Public Key, and other attributes.
- Certificate Authority (CA): transforms the certRequest into an X.509 certificate after authentication and verification

- PKCS#10 certRequest: contains digitally signed certificate request information including Subject Name and Public Key
- Simple Certificate Enrollment Protocol (SCEP) client: provides functionality to wrap PKCS#10 in PKCS#7 Signed Data message type

Workflow

Your device certificate enrollment process works through these stages:

1. The Lightweight Access Point (LAP) and controller create a PKCS#10 certificate request containing the Subject Name, Public Key, and other attributes to be included in the X.509 certificate, and digitally sign it with the Private Key of the requester.
2. The PKCS#10 is wrapped in a PKCS#7 Signed Data message type as part of the Simple Certificate Enrollment Protocol (SCEP) client functionality, and the PKCSReq message is sent to the controller.
3. The Certificate Authority (CA) receives the PKCS#10 certRequest and authenticates the identity of the requester and verifies that the request is unaltered.
4. The Certificate Authority (CA) transforms the certRequest into an X.509 certificate.

Result

Upon successful enrollment operation, both the Certificate Authority (CA) and device certificates are available on the controller.

How certificate provisioning works on lightweight APs

Certificate provisioning enables lightweight APs (LAPs) to obtain new signed X.509 certificates while operating in CAPWAP mode. This process ensures secure communications between APs and controllers using updated security credentials.

Summary

The key components involved in certificate provisioning on lightweight APs are:

- Lightweight Access Point (LAP): Initiates certificate requests and receives new certificates while in CAPWAP mode
- Controller: Acts as a CA proxy to help obtain signed certificates from the Certificate Authority for the LAP
- Certificate Authority (CA): Signs and validates certificate requests to issue new X.509 certificates
- LWAPP payloads: Transport mechanism for certificate requests and responses between LAP and controller

Workflow

These stages describe how certificate provisioning works on lightweight APs:

1. The LAP sends a certificate request to the controller to obtain a new signed X.509 certificate.

   The controller acts as a CA proxy and helps obtain the certificate request signed by the CA for the LAP.

2. The certificate requests and certificate responses are transmitted between the LAP and controller using LWAPP payloads.
3. Both the LSC CA and LAP device certificates are installed in the LAP, and the system reboots automatically.
4. When the system comes up after reboot, the AP sends the LSC device certificate to the controller as part of the JOIN Request because it is configured to use LSCs.

   As part of the JOIN Response, the controller sends the new device certificate and validates the inbound LAP certificate with the new CA root certificate.

What's next

To configure, authorize, and manage certificate enrollment with the existing PKI infrastructure for controller and AP, you need to use the LSC provisioning functionality.

Restrictions for locally significant certificates

- The locally significant certificate (LSC) workflow differs in FIPS and WLANCC mode. Your Certificate Authority (CA) server must support Enrollment over Secure Transport (EST) protocol. Your CA server must also issue EC certificates in FIPS and WLANCC mode.
- The Elliptic Curve Digital Signature Algorithm (ECDSA) cipher works only if both your AP and controller have EC certificates provisioned with LSC.
- EC certificates (LSC-EC) can be provisioned only if CA server supports EST (and not SCEP).
- You must configure FIPS and CC security modes to provision EC certificates.

Provisioning locally significant certificates

Configure RSA key for PKI trustpoint

Generate RSA cryptographic keys for PKI trustpoint authentication and secure communications.

RSA keys are essential components of PKI infrastructure, providing the cryptographic foundation for certificate-based authentication and encryption on wireless controllers.

Before you begin

Ensure the device is in a stable operational state and has sufficient processing resources for key generation.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Generate RSA key for PKI trustpoint.

Example:

```
Device(config)# crypto key generate rsa exportable general-keys modulus key-size label
RSA-key-label
```

Example:

```
Device(config)# crypto key generate rsa exportable
general-keys modulus 2048 label lsc-tp
```

exportable is an optional keyword. You may or may not want to configure an exportable-key. If selected, you can export the key out of the box, if required.

- *key-size*: Size of the key modulus. The valid range is from 2048 to 4096.
- *RSA-key-label*: RSA key pair label.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The RSA key is successfully generated and labeled for use with PKI trustpoint configurations.

Configure PKI trustpoint parameters

Configure a PKI trustpoint with RSA key parameters for certificate enrollment and authentication.

PKI trustpoints define the relationship between the device and a Certificate Authority (CA), enabling secure certificate-based authentication and encryption.

Before you begin

Ensure the CA server is accessible and configured to accept certificate requests.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new trustpoint for an external CA server.

Example:

```
Device(config)# crypto pki trustpoint trustpoint-name
```

Example:

```
Device(config)# crypto pki trustpoint microsoft-ca
```

The *trustpoint-name* refers to the trustpoint name.

Step 3 Specify the URL of the CA on which your router should send certificate requests.

Example:

```
Device(ca-trustpoint)# enrollment url HTTP-URL
```

Example:

```
Device(ca-trustpoint)# enrollment url http://CA_server/certsrv/mscep/mscep.dll
```

HTTP-URL: URL of the file system where your router 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. For more enrollment method options, see the enrollment url (ca-trustpoint) command page.

Step 4 Create subject name parameters for the trustpoint.

Example:

```
Device(ca-trustpoint)# subject-name subject-name
```

Example:

```
Device(ca-trustpoint)# subject-name C=IN, ST=KA, L=Bengaluru, O=Cisco,
CN=eagle-eye/emailAddress=support@abc.com
```

Step 5 Map RSA key with that of the trustpoint.

Example:

```
Device(ca-trustpoint)# rsakeypair RSA-key key-size
```

Example:

```
Device(ca-trustpoint)# rsakeypair ewlc-tp1
```

- *RSA-key*: RSA key pair label.
- *key-size*: Signature key length. Range is from 360 to 4096.

Step 6 Configure revocation checking method.

Example:

```
Device(ca-trustpoint)# revocation {crl | none | ocsp}
```

Example:

```
Device(ca-trustpoint)# revocation none
```

Step 7 Return to privileged EXEC mode.

Example:

```
Device(ca-trustpoint)# end
```

The PKI trustpoint is now configured with the specified parameters and ready for certificate enrollment operations.

Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI)

Use this procedure to add and authenticate a Public Key Infrastructure (PKI) trustpoint to enable certificate-based security for your device.

Before you begin

Obtain the enrollment URL and certificate authority information from your network administrator.

Procedure

Step 1 Choose **Configuration** > **Security** > **Public Key Infrastructure (PKI) Management**.

Step 2 In the **Public Key Infrastructure (PKI) Management** window, click the **Trustpoints** tab.

Step 3 In the **Add Trustpoint** dialog box, provide this information:

a) In the **Label** field, enter the RSA key label.
b) In the **Enrollment URL** field, enter the enrollment URL.
c) Check the **Authenticate** check box to authenticate the Public Certificate from the enrollment URL.
d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organization**, **Domain Name**, and **Email Address**.
e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list.
f) Check the **Enroll Trustpoint** check box.
g) In the **Password** field, enter the password.
h) In the **Re-Enter Password** field, confirm the password.
i) Click **Apply to Device**.

The new trustpoint is added to the trustpoint name list.

The new trustpoint is now listed in the trustpoint name list and is ready for use.

What to do next

After enrolling the trustpoint, configure it for use with your security policies or certificate-based authentication features.

Authenticate and enroll the PKI trustpoint with CA server (CLI)

Establish secure communication between the device and Certificate Authority by authenticating and enrolling a PKI trustpoint.

PKI trustpoints enable secure certificate-based authentication. The authentication process fetches the CA certificate, while enrollment generates and submits a certificate request to obtain a client certificate.

Before you begin

Ensure the PKI trustpoint is configured and the CA server is accessible.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Fetch the CA certificate.

Example:

```
Device(config)# crypto pki authenticate trustpoint-name
```

Example:

```
Device(config)# crypto pki authenticate microsoft-ca
```

Step 3 Accept the CA certificate.

Example:

```
Device(config)# % Do you accept this certificate? [yes/no]:
yes Trustpoint CA certificate accepted.
```

Step 4 Enroll the client certificate.

Example:

```
Device(config)# crypto pki enroll trustpoint-name
```

Example:

```
Device(config)# crypto pki enroll microsoft-ca
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally
provide this password to the CA Administrator in order to
revoke your certificate. For security reasons your password
will not be saved in the configuration.
Please make a note of it.
```

Step 5 Enter a challenge password to the CA server.

Example:

```
Device(config)# password
```

Example:

```
Device(config)# abcd123
```

Step 6 Re-enter the challenge password to the CA server.

Example:

```
Device(config)# password
```

Example:

```
Device(config)# abcd123
```

Step 7 Include the router serial number in the subject name.

Example:

```
Device(config)# % Include the router serial number
in the subject name? [yes/no]: yes
```

Step 8 Exclude IP address from the subject name.

Example:

```
Device(config)# % Include an IP address
in the subject name? [no]: no
```

Step 9 Request certificate from CA.

Example:

```
Device(config)#
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
```

```
% The 'show crypto pki certificate verbose
client' command will show the fingerprint.
```

Step 10 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The PKI trustpoint is authenticated and enrolled with the CA server, enabling secure certificate-based communication.

Configure AP join attempts with LSC certificate

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **All Access Points** window, click the LSC Provision name.

Step 3 From the **Status** drop-down list, choose a status to enable LSC.

Step 4 From the **Trustpoint Name** drop-down list, choose the trustpoint.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that will be permitted.

Step 6 Click **Apply**.

Configure AP join attempts with LSC certificate (CLI)

Configure the maximum number of AP join failure attempts with a newly provisioned LSC certificate to ensure proper fallback to MIC when needed.

When APs attempt to join with LSC certificates and exceed the configured failure limit, they automatically fall back to using their Manufacturer Installed Certificate (MIC) for authentication.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the maximum number of AP join failure attempts with the newly provisioned LSC certificate.

Example:

```
Device(config)# ap lsc-provision join-attempt number-of-attempts
```

Example:

```
Device(config)# ap lsc-provision join-attempt 10
```

When the number of AP joins exceed the specified limit, AP joins back with the Manufacturer Installed Certificate (MIC).

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The AP join attempt limit with LSC certificate is now configured. APs will fall back to MIC authentication after exceeding the specified number of failed join attempts.

Configure subject-name parameters in LSC certificate

Configure the subject-name parameters that will be included in LSC certificates generated by access points.

Subject-name parameters define the identity information included in Local Significant Certificate (LSC) requests generated by access points. These parameters include geographical and organizational details that identify the certificate holder.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the attributes to be included in the subject-name parameter of the certificate request generated by an AP.

Example:

```
Device(config)# ap lsc-provision subject-name-parameter
country country-str state state-str city city-str domain domain-str org
org-str email-address email-addr-str
```

Example:

```
Device(config)# ap lsc-provision subject-name-parameter
country India state Karnataka city Bangalore domain domain1 org
Right email-address adc@gfe.com
```

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The subject-name parameters are now configured for LSC certificate generation on access points.

Configure key size for LSC certificate

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the size of keys to be generated for the LSC on AP.

Example:

```
Device(config)# ap lsc-provision key-size key-size
```

Example:

```
Device(config)# ap lsc-provision key-size 2048
```

Valid key size options are 2048, 3072, or 4096 bits.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Configure trustpoint for LSC provisioning on an AP

Establish a trustpoint for Local Significant Certificate (LSC) provisioning to enable secure communication between access points and the controller.

LSC provisioning allows access points to obtain certificates from a designated trustpoint, enhancing network security through certificate-based authentication.

Before you begin

Ensure the trustpoint is properly configured and accessible on the network.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the trustpoint with which the LSC is provisioned to an AP.

Example:

```
Device(config)# ap lsc-provision trustpoint tp-name
```

Example:

```
Device(config)# ap lsc-provision trustpoint microsoft-ca
```

The *tp-name* is the trustpoint name.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The trustpoint for LSC provisioning is now configured and ready to provision certificates to access points.

Configure an AP LSC provision list (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **All Access Points** window, click the corresponding LSC Provision name.

Step 3 From the **Status** drop-down list, choose a status to enable LSC.

Step 4 From the **Trustpoint Name** drop-down list, choose a trustpoint.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that are allowed.

Step 6 From the **Key Size** drop-down list, choose a key.

Step 7 In the **Edit AP Join Profile** window, click the **CAPWAP** tab.

Step 8 In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains AP details.

Step 9 Click **Upload File**.

Step 10 In the **AP MAC Address** field, enter the AP MAC address. and add them. (The APs added to the provision list are displayed in the **APs in provision List** .)

Step 11 In the **Subject Name Parameters** section, enter the following details:

- **Country**
- **State**
- **City**
- **Organization**
- **Department**
- **Email Address**

Step 12 Click **Apply**.

Configure an AP LSC provision list (CLI)

Add access points to the LSC provision list for certificate-based authentication.

The LSC provision list allows you to specify which access points can use locally significant certificates for authentication with the wireless controller.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Add the AP to the LSC provision list.

Example:

```
Device(config)# ap lsc-provision mac-address mac-address
```

Example:

```
Device(config)# ap lsc-provision
mac-address 001b.3400.02f0
```

Note
You can provision a list of APs using the **ap lsc-provision provision-list** command.

(Or)

You can provision all the APs using the **ap lsc-provision** command.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The AP is now added to the LSC provision list and can use locally significant certificates for authentication.

Configure LSC provisioning for all the APs (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Access Points** window, expand the **LSC Provision** section.

Step 3 Set **Status** to **Enabled** state.

Note
If you set **Status** to **Provision List**, LSC provisioning will be configured only for APs that are a part of the provision list.

Step 4 From the **Trustpoint Name** drop-down list, choose the appropriate trustpoint for all APs.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that the APs can make to join the embedded wireless controller.

Step 6 From the **Key Size** drop-down list, choose the appropriate key size of the certificate:

- **2048**
- **3072**
- **4096**

Step 7 In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains the AP details.

Step 8 Click **Upload File**.

Step 9 In the **AP MAC Address** field, enter the AP MAC address. (The APs that are added to the provision list are displayed in the **APs in Provision List** section.)

Step 10 In the **Subject Name Parameters** section, enter the following details:

a. **Country**

b. **State**

c. **City**

d. **Organization**

e. **Department**

f. **Email Address**

Step 11 Click **Apply**.

Configure LSC provisioning for all APs (CLI)

Enable LSC (Local Significant Certificate) provisioning across all access points to enhance security through automatic certificate management.

LSC provisioning allows access points to automatically obtain and manage security certificates, reducing manual certificate deployment overhead while maintaining secure wireless communications.

Before you begin

Ensure you have administrative access to the wireless controller and that all APs are properly registered with the controller.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable LSC provisioning for all APs.

Example:

```
Device(config)# ap lsc-provision
```

By default, LSC provisioning is disabled for all APs.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

LSC provisioning is now enabled for all access points, allowing them to automatically obtain and manage security certificates.

Configure LSC provisioning for the APs in the provision list

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable LSC provisioning for a set of APs configured in the provision list.

Example:

```
Device(config)# ap lsc-provision provision-list
```

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Verify LSC configuration

Use show commands to verify Local Security Certificate (LSC) configuration. View wireless management trustpoint details and Local Security Certificate (LSC) provision-related configuration for your access points.

To view the wireless management trustpoint details, use this command:

```
Device# show wireless management trustpoint

Trustpoint  Name : microsoft-ca
Certificate Info : Available
Certificate Type : LSC
Certificate Hash : 9e5623adba5307facf778e6ea2f5082877ea4beb
 Private key Info : Available
```

To view the Local Security Certificate (LSC) provision-related configuration details for an AP, use this command:

```
Device# show ap lsc-provision summary

AP LSC-provisioning : Disabled
Trustpoint used for LSC-provisioning : microsoft-ca
```

```
LSC Revert Count in AP reboots : 10

AP LSC Parameters :
Country : IN
State : KA
City : BLR
Orgn : ABC
Dept : ABC
Email : support@abc.com
Key Size : 2048

AP LSC-provision List : Enabled
Total number of APs in provision list: 3

Mac Address
-----------
0038.df24.5fd0
2c5a.0f22.d4ca
e4c7.22cd.b74f
```

Configure management trustpoint to LSC

Procedure

Step 1 Choose **Administration** > **Management** > **HTTP/HTTPS**.

Step 2 In the **HTTP Trust Point Configuration** section, set **Enable Trust Point** to the **Enabled** state.

Step 3 From the **Trust Points** drop-down list, choose the appropriate trustpoint.

Step 4 Save the configuration.

Configure management trustpoint to LSC (CLI)

Configure management trustpoint to enable Local Significant Certificate (LSC) authentication for access points.

After LSC provisioning, the APs will automatically reboot and join at the LSC mode after bootup. Similarly, if you remove the AP LSC provisioning, the APs reboot and join at non-LSC mode.

In EWC, the internal APs will not automatically reboot. You should manually reboot the internal AP to make it work in LSC and non-LSC mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the management trustpoint to LSC.

Example:

```
Device(config)# wireless management trustpoint trustpoint-name
```

Example:

```
Device(config)# wireless management trustpoint microsoft-ca
```

The internal AP will not able to join before a reload, so follow the steps given below to reload the internal AP.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 4 Save the configuration.

Example:

```
Device# copy running-config startup-config
```

Step 5 Reload the internal AP.

Example:

```
Device# wireless ewc-ap ap reload
```

This will also reload the controller on the AP.

The management trustpoint is now configured for LSC authentication, and access points will join using LSC mode after rebooting.

AP certificate renewal (LSC)

AP certificate renewal (LSC)

AP certificate renewal (LSC) is a feature that

- allows the renewal of Locally Significant Certificates (LSCs) for Access Points before their expiration
- ensures that the APs can continue to provide seamless services by renewing their certificates in a timely manner
- enables the controller to orchestrate this process to minimize service disruption. The process focuses on automatic renewal mechanisms that do not require manual intervention.

LSC provisioning and renewal methods

This feature supports LSC provisioning for Access Points from two sources. You can provision from the controller through proxy-enrollment or from the Cisco Catalyst Center through secure AP onboarding. These certificates are crucial for your Access Points because they authenticate network communications. This renewal process is essential. Without a valid certificate, Access Points may lose network connectivity and affect network services.

The controller uses these methods to renew certificates:

- **Gradual Succession**: A method where Access Points are renewed one at a time, which is suitable when the number of Access Points that are due for renewal is under a certain threshold.
- **Staggered Rollover**: Access Points are renewed in a staggered fashion, minimizing the impact on radio coverage when the number of Access Points is larger.
- **One-Shot Renewal**: All affected Access Points are renewed simultaneously, which can be disruptive and should be used as a last resort.

Benefits of AP certificate renewal (LSC)

- **Automated Renewal Process**: Reduces the need for manual certificate updates, ensuring continuous service.
- **Seamless Network Operations**: Minimizes network downtime by renewing certificates before they expire.
- **Enhanced Security**: Maintains a high level of security by ensuring that all Access Points have valid certificates.
- **Scalable Management**: Supports staggered AP renewal to reduce impact on network operations.

Requirements for AP certificate renewal (LSC)

Configure the controller and access points to use locally significant certificates before you proceed with certificate renewal.

- Provision the locally significant certificates through the controller (proxy-enrollment) or Cisco Catalyst Center-PNP workflows.
- Define the maintenance windows if scheduling is required for renewal.

Restrictions for AP certificate renewal (LSC)

These restrictions for access point certificate renewal (Local Significant Certificate) are:

- Only one Local Significant Certificate is supported per access point.
- Enable either staggered or one-shot renewal, not both.
- Local Significant Certificate renewal applies to Cisco Wave 2 access points and newer models; older access points do not support this feature.
- The staggered algorithm does not apply to access points in bridge mode. All Local Significant Certificate renewals for bridge mode access points occur at the same time.

Configure AP certificate auto-renewal (LSC) (GUI)

Follow these steps to configure AP certificate auto-renewal (LSC):

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points** > **LSC Provision**.

Step 2 In the **LSC Renewal** section, check the **LSC Auto-Renewal** check box and follow these steps:

a. Enter a value for due time. The valid range is 5 to 60.

b. From the **Renewal Schedule** drop-down list, select an existing calendar profile schedule or click the icon to create a new calendar profile. For more information about creating a calendar profile, see the **Configuring a Calendar Profile (GUI)** and **Configuring a Calendar Profile (CLI)**.

c. Use the slider to enable **Syslog** messages for the LSC renewal updates.

d. From the **Renewal Mode** drop-down list, choose either *One Shot* or *Staggered*.

- If you choose One Shot, all impacted AP LSCs renew in a single iteration.
- If you choose Staggered, the system staggers the LSC renewal configuration for the APs and follow these steps:
  - Enter a value for the iteration expiry to set the expiry time for each iteration.
  - From the **AP LSC Renewal per Iteration** drop-down list, choose one of the following options to configure the percentage of APs to be renewed per iteration: 5%, 15%, or 25%.
  - Enable the **Client Steering** slider to move clients attached to APs undergoing renewal to other APs.
  - Enable the **Client Deauthentication** slider to deauthenticate clients associated with the AP before it starts upgrading.
  - From the **Accounting Percentage** drop-down list, select the percentage of APs that should join the controller after each iteration to consider the renewal successful.
  - Use the slider to configure **Accounting Action**. If you select **Terminate**, the renewal stops if the configured percentage of APs does not join the controller. If you select **Ignore**, the renewal continues regardless.

Step 3 Click **Apply**.

Configure AP certificate renewal (LSC) (CLI)

Follow these steps to configure AP certificate renewal (LSC) in the controller:

Procedure

Step 1 Enter configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure LSC provisioning on an AP.

Example:

```
Device(config)# ap lsc-provision
```

Step 3 Configure AP certificate renewal.

Example:

```
Device(config)# ap lsc-provision renew
```

Step 4 Renew AP LSCs automatically within the specified number of days.

Example:

```
Device(config)# ap lsc-provision renew due-time number-of-days
```

The valid range is from 5 to 60 days, with a default value of 30 days.

Step 5 Renew all AP LSCs in a single iteration.

Example:

```
Device(config)# ap lsc-provision renew one-shot
```

Note
The one-shot AP LSC renewal feature is disabled by default.

Step 6 Renew AP LSCs automatically in a staggered manner.

Example:

```
Device(config)# ap lsc-provision renew staggered timeout iteration-timeout-in-mins
```

Staggered AP LSC renewal is enabled by default.

Note
You cannot use one-shot and staggered renewal methods at the same time.

iteration-timeout-in-mins: The valid range is from 5 to 60 minutes.

Step 7 Configure time schedule for auto-renewal within a maintenance window.

Example:

```
Device(config)# ap lsc-provision renew scheduler calendar-profile
```

No scheduler is enabled by default.

When no scheduler is applied, the device auto-renews AP LSCs at 24-hour intervals and is triggered daily at 12:00 AM local time.

Note
You must configure the calendar-profile definition beforehand. For more information, see **Configuring a Calendar Profile (CLI)**.

You have successfully configured AP certificate renewal (LSC).

Configure forced renewal of AP certificate (LSC) for individual or multiple APs (GUI)

Follow these steps to configure forced renewal of AP certificate (LSC) for individual or multiple APs via GUI:

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Bulk AP Provisioning**.

Step 2 On the **Select Parameters** page, check the **Renew LSC** check box to renew the LSC for all selected APs.

Step 3 Click **Next**.

Configure forced renewal of AP Certificate (LSC) based on site tag (GUI)

Before you begin

Follow these steps to configure force renewal of AP certificate (LSC) based on site tag:

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 In the **Site** tab, check the check box for the required site tag.

Step 3 Click **Renew LSC** and follow these steps:

a. Enter the due time in days. The value must be between 1 and 60.

b. Click **Apply to Device**.

Configure forced renewal of AP Certificate (LSC) based on site tag (CLI)

Follow these steps to configure the forced renewal of an AP certificate (LSC) based on a site tag using the CLI:

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Renew a specific site-tag AP that is set to expire within the specified time.

Example:

```
Device# ap lsc renew due-time number-of-days site-tag site-tag-name
```

number-of-days: The valid range is from 1 to 60 days.

The configuration of the forced renewal of an AP Certificate (LSC) based on a site tag is complete.

Configure forced renewal of all APs (GUI)

Follow these steps to configure forced renewal of all APs (GUI):

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 From the **Select an Action** drop-down list, choose **Renew LSC**.

The **Renew LSC** pop-up window is displayed.

Step 3 Enter a due time value in minutes between 5 and 60.

Step 4 Click **Apply to Device**.

Verify AP certificate renewal (LSC)

Lists the various show commands and their outputs during the verification of the AP Certificate Renewal (LSC).

Verify LSC configuration

To verify LSC provision-related configuration details for an AP, use this command:

```
Device# show ap lsc-provision info
AP name     Ethernet MAC     Radio MAC      LSC authentication  LSC workflow    Cert expiry
   Last renew attempt  Last renew failure
=============================================================================================
AP-abc      1111.2222.3333  aaaa.bbbb.ccc0  Port-802.1x         PNP Onboarding  <time-stamp>
  <time-stamp>/NA     reason-code
AP-123      1234.5678.9012  dddd.eeee.fff0  DTLS                SCEP proxy      <time-stamp>
  <time-stamp>/NA     reason-code
AP-xyz      1010.1010.1010  abcd.efab.cde0  DTLS, Port-802.1x   SCEP proxy      <time-stamp>
  <time-stamp>/NA     reason-code
                                      0.0.0.0                          372            64
              AGL          Enabled
```

Verify staggered LSC renewal upgrade configuration

To verify progress and status of staggered LSC renewal upgrade, use this command:

```
Device# show ap upgrade
```

```
Status: In progress
From version:
To version: Unknown

Started at: 10/23/2025 08:16:01 IST
Configured percentage: 15
Percentage complete: 0
Expected time of completion: 10/23/2025 08:24:01 IST

Client steering: Enabled
Client de-authentication: Enabled
Accounting percentage: 90%
Iteration expiry time: 15 minutes

Progress Report
---------------
Iterations
----------
Iteration              Start time                          End time                    AP
count
-------------------------------------------------------------------------------------------------
0                      10/23/2025 08:16:01 IST             10/23/2025 08:16:01 IST     0

1                      10/23/2025 08:16:01 IST             10/23/2025 08:18:05 IST     1

2                      10/23/2025 08:18:05 IST             ONGOING                     0


Upgraded
--------
Number of APs: 0
AP Name                           Radio MAC               Iteration       Status
    Site
-------------------------------------------------------------------------------------------------------

In Progress
-----------
Number of APs: 1
AP Name                           Radio MAC
-------------------------------------------------
1815W                             00be.7517.d980

Remaining
---------
Number of APs: 6
AP Name                           Radio MAC
-------------------------------------------------
9166-1                            10f9.20fd.5400
non-9120-1                        488b.0a47.d8c0
9136_1                            687d.b45f.1610
9120-Dev                          a00f.3704.f3a0
9162-antenna                      ecf4.0c0e.9920
9176+alpine                       ecf4.0c92.bac0

APs not handled by Rolling AP Upgrade
-------------------------------------
AP Name                           Radio MAC               Status          Reason for not
handling by Rolling AP Upgrade
-------------------------------------------------------------------------------------------------------
```

MIC and LSC APs joining the controller

Support for MIC and LSC APs joining the controller

Support for MIC and LSC access points joining the controller is an authorization policy feature that

- allows MIC APs to join LSC-deployed controllers without requiring LSC provisioning
- enables LSC and MIC APs to coexist on the same controller simultaneously, and
- eliminates the need to provision LSC on MIC APs before moving them to LSC-deployed controllers.

Authorization policy behavior changes

The authorization policy behavior differs between software releases:

- **Cisco IOS XE Bengaluru 17.4.1 and earlier**: If your APs have default certificates (Manufacturing Installed Certificates [MIC] or Secure Unique Device Identifier [SUDI]), they cannot join an LSC-deployed controller, where the management certificate of the controller is an LSC. To resolve this issue, provision LSC on your APs. Use the provisioning controller before you move them to the LSC-deployed controller.
- **Cisco IOS XE Bengaluru 17.5.1 onwards**: The new authorization policy configuration allows MIC APs to join the LSC-deployed controller, so that LSC APs and MIC APs can coexist on the controller.

Recommendations and limitations

When you configure the CA server with manual enrollment to accept certificate signing request (CSR), the controller monitors for the CA server response. If the CA server does not respond for 10 minutes, the fallback mode activates.

- Cisco Wave 2 APs regenerate CSR, and a fresh CSR is sent to the CA server.
- Cisco IOS APs restart, and then Cisco IOS APs send a fresh CSR, which is in turn sent to the CA server.

The locally significant certificate (LSC) feature on the controller does not work with password challenge authentication. Therefore, for LSC to work, you must disable password challenge on the CA server.

If you are using Microsoft CA, use Windows Server 2012 or later as the CA server.

MIC and LSC access point authentication

Summary

The Manufacturer Installed Certificate (MIC) and Local Significant Certificate (LSC) access point authentication configuration involves these stages:

- Controller: Manages authentication policies and trustpoint configuration for secure access point connections
- Access points: Require certificate-based authentication to join the controller
- LSC trustpoint: Provides local significant certificate authentication mechanism

- AP certificate policy: Defines authentication requirements and allowed access point lists

Workflow

The key components involved in the MIC and LSC access point authentication configuration are:

1. The network administrator configures the LSC trustpoint on the LSC deployed controller using CLI.
   Configure LSC on the controller (CLI), on page 527
2. The network administrator enables the AP certificate policy on access points using CLI.
   Enable the AP certificate policy on the APs (CLI), on page 528
3. The network administrator configures the AP policy certificate using the GUI interface.
   Configure the AP policy certificate (GUI), on page 529
4. The network administrator configures the allowed list of APs to join the controller using CLI.
   Configure the allowed list of APs to join the controller (CLI), on page 529

Result

Access points are properly authenticated and authorized to join the controller with secure certificate-based authentication enabled.

Configure LSC on the controller (CLI)

The server certificate used by the controller for CAPWAP-DTLS (Control and Provisioning of Wireless Access Points - Datagram Transport Layer Security) is based on the following configuration.

Before you begin

- Ensure that you enable LSC by setting the appropriate trustpoints for the following wireless management services:
  - AP join process: CAPWAP DTLS server certificate
  - Mobility connections: Mobility DTLS certificate
  - NMSP (Network Mobility Services Protocol) and CMX (Connected Mobile Experiences) connections: NMSP TLS certificate

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the LSC trustpoint in the LSC-deployed controller.

Example:

```
Device(config)# wireless management trustpoint trustpoint-name
```

Enable the AP certificate policy on the APs (CLI)

Enable AP certificate policy during CAPWAP-DTLS handshake to allow MIC APs to join the controller.

- If the management trustpoint is a locally significant certificate (LSC), by default, manufacturing installed certificate (MIC) APs fail to join the controller. This configuration acts as an enable or disable configuration knob that allows MIC APs to join the controller.
- This configuration is a controller authorization to allow APs to join MIC at the time of Datagram Transport Layer Security (DTLS) handshake.

To prevent manufacturing installed certificate (MIC) expiry failures, ensure that you configure a policy, as shown here:

- Create a certificate map and add the rules:

```
configure terminal
crypto pki certificate map map1 1
issuer-name co Cisco Manufacturing CA
```

Note You can add multiple rules and filters under the same map. The rule mentioned in the example above specifies that any certificate whose issuer-name contains *Cisco Manufacturing CA* (case insensitive) is selected under this map.

- Use the certificate map under the trustpool policy:

```
configure terminal
crypto pki trustpool policy
match certificate map1 allow expired-certificate
```

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the trustpoint name for the controller certificate chain.

Example:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint trustpoint-name
```

Note
The allow-mic-ap trustpoint command is required only for the virtual controller (Cisco Catalyst 9800-CL Wireless Controller for Cloud). In all the other appliance controller platforms, the default certificate is selected. This default certificate is manufacturer-installed SUDI.

Step 3 Enable the AP certificate policy during CAPWAP-DTLS handshake.

Example:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap
```

Step 4 Enable the AP certificate policy as MIC.

Example:

```
Device(config)# ap auth-list ap-cert-policy mac-address H.H.H policy-type mic
```

Example:

```
Device(config)# ap auth-list ap-cert-policy mac-address 1111.1111.1111 policy-type mic
```

The AP certificate policy is now enabled and configured for MIC APs to join the controller during CAPWAP-DTLS handshake.

Configure the AP policy certificate (GUI)

Procedure

Step 1 Choose **Configuration > Wireless > Access Points**

Step 2 In the **All Access Points** window, click **AP Certificate Policy**.

Step 3 In the **AP Policy Certificate** window, complete the following actions:

a) Click the **Authorize APs joining with MIC** toggle button to enable AP authorization.

b) From the **Trustpoint Name** drop-down list, choose the required trustpoint.

c) Click **Add MAC or Serial Number** to add a MAC address or a serial number manually or through a .csv file.
The **Add MAC or Serial Number** window is displayed.

d) Click the **AP Authlist Type** and enter the MAC address or the serial number. Upload the .csv file or enter the MAC address in the list box.
The newly added MAC address and serial numbers are displayed under **List of MAC Address and Serial Numbers**.

e) Click **Apply**.

The AP certificate policy is added to the **AP Inventory** window.

Note
To add a new AP with MIC, perform Step 1 to Step 3 described in Configuring the AP Policy Certificate (GUI) section. To add a new AP with LSC, perform the procedure described in the Configuring AP LSC Provision List (GUI) and Step 1 to Step 3 in the Configuring the AP Policy Certificate (GUI) section.

Configure the allowed list of APs to join the controller (CLI)

Create an allowed list of access points that can join the controller based on MAC address or serial number authentication.

The allowed list of APs can either be populated based on the Ethernet MAC address or based on the serial number of the APs.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the AP certificate policy based on the Ethernet MAC address or based on the assembly serial number of the AP.

Example:

```
Device(config)# ap auth-list ap-cert-policy mac-address AP-Ethernet-MAC-address policy-type
 mic
```

Example:

```
Device(config)# ap auth-list ap-cert-policy serial-number AP-serial-number policy-type mic
```

Example:

```
Device(config)# ap auth-list ap-cert-policy mac-address 00b0.e192.0d98 policy-type mic
```

The AP certificate policy is configured and the specified access points are added to the allowed list for joining the controller.

Verifying the configuration status

To verify if the APs have been authorized by the AP certificate policy, use this command:

```
Device# show ap auth-list ap-cert-policy
Authorize APs joining with MIC : ENABLED
MIC AP policy trustpoint
Name : CISCO_IDEVID_SUDI
Certificate status : Available
Certificate Type : MIC
Certificate Hash : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

To verify the AP certificate policy on the MAC address and the serial number of the AP, use these commands:

```
Device# show ap auth-list ap-cert-policy mac-address
MAC address      AP cert policy
---------------------------------
1111.2222.3333   MIC
```

```
Device# show ap auth-list ap-cert-policy serial-number
Serial number    AP cert policy
---------------------------------
F1234567890      MIC
```

Note If you set an invalid trustpoint (not SSC), the **allow-mic-ap policy** does not enable. If you set an invalid trustpoint, this error displays on the console:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint lsc-root-tp
Dec 18 07:38:29.944: %CERT_MGR_ERRMSG-3-CERT_MGR_GENERAL_ERR: Chassis 1 R0/0: wncd: General
 error: MIC AP Policy trustpoint: 'lsc-root-tp' cert-chain type is LSC, It must be either
MIC or vWLC-SSC
```

LSC fallback APs

LSC fallback APs

LSC fallback is a state that

- occurs when an AP configured with LSC for CAPWAP fails to establish DTLS connection after reaching the maximum number of retry attempts
- causes the AP to revert to its default certificate (MIC) for CAPWAP authentication, and
- provides a backup authentication method when LSC-based CAPWAP connection attempts fail.

LSC fallback process

When you configure an AP with LSC for CAPWAP but it fails to establish a DTLS connection, the AP reboots. The AP then retries for a certain number of times. For information on how to configure an AP with LSC, see Configure AP join attempts with LSC certificate (CLI), on page 511.

The AP falls back to its default certificate (MIC) for CAPWAP after maximum number of failures. The AP fallback state is referred to as the LSC fallback.

Note MIC is also known as SUDI certificate.

Troubleshoot LSC fallback state

This task enables you to identify access points in LSC fallback state and understand the recovery process when APs fail to establish proper LSC connections with the controller.

When an AP in LSC fallback state joins the controller, it generates specific syslog messages and is held in a misconfigured state. The AP does not broadcast WLAN or SSID configurations in this state, which permits the admin to examine the reason for previous failures and recover APs.

Procedure

Step 1 Identify the syslog message generated when an AP in LSC fallback state joins the controller.

Example:

```
Jun 15 23:24:14.836: %APMGR_TRACE_MESSAGE-3-WLC_GEN_ERR: Chassis 1 R0/0: wncd: Error
in AP: 'AP2c5a.0f70.84dc' with address 70db.9888.cc20 is joined with MIC, while configuration

requires LSC. No WLANs will be pushed.
```

The controller allows such an AP to be joined with MIC (when AP certificate policy allows it) and AP is held in misconfigured state.

Note
The AP does not broadcast WLAN or SSID configurations in such state. This permits the admin to examine the reason for previous failures and recover APs.

Step 2 Use **show wireless summary** to identify LSC fallback APs.

Example:

```
Device# show wireless summary
…
Access Point Summary
…
DTLS LSC fallback APs     20 (No WLANs will be pushed to these APs)
…
For more information on DTLS LSC fallback APs,
    execute 'wireless config validate' and look for reported errors in
    'show wireless config validation status' CLI output.

Use 'show ap config general | inc AP Name | LSC fallback' to list DTLS LSC fallback APs.
Examine LSC fallback reasons / DTLS handshake failures with LSC then
    issue 'ap lsc dtls-fallback clear-certificate / clear-flag' to recover APs
```

You have successfully identified LSC fallback APs and understand the commands available for further investigation and recovery.

Recover from LSC fallback

Recover APs from LSC fallback state and restore normal LSC functionality for CAPWAP DTLS connections.

When LSC fallback occurs, APs may require recovery actions to restore normal operation. Recovery options depend on whether you want to retain the LSC certificate or clear it completely.

Procedure

Choose one of the following recovery methods based on your requirements:

- To clear the LSC fallback flag and retain the LSC certificate, use the **ap lsc dtls-fallback clear-flag** command to instruct the AP to reload.

  Note
  The AP reuses the LSC for CAPWAP DTLS connection post the reload.

- To clear the LSC certificate completely, use the **ap lsc dtls-fallback clear-certificate** command to clear LSC and instruct the AP to reload.

Note
The AP uses MIC for CAPWAP-DTLS post the reload. If LSC is used for Dot1x port authentication then further recovery is needed on switch port for AP authentication.

Note
- The **ap lsc dtls-fallback clear-flag** command is sufficient to retain LSC on AP. Both **ap lsc dtls-fallback clear-flag** and **ap lsc dtls-fallback clear-certificate** commands are not required at the same time.
- APs must be in connected state when issuing the recovery command. You will need to reissue the command, if any **LSC fallback** AP joins afterwards.

The access point reloads and recovers from the LSC fallback state according to the selected recovery method.

Configuring controller self-signed certificate for wireless AP join

Use cases for controller self-signed certificate for wireless AP join

You need controller self-signed certificates in specific circumstances to enable wireless APs to join the controller successfully.

Controller self-signed certificate use cases

These situations require controller self-signed certificates:

- The Cisco Catalyst 9800-CL platform is not equipped with manufacturer-installed SUDI certificates. Configure self-signed certificates on these controllers.
- Some APs running earlier software and using a Manufacturer Installed Certificate (MIC) issued by a SHA1 Cisco Trusted CA cannot join controllers with SHA2 SUDI certificates. During the CAPWAP join process, these APs display a bad certificate error and fail the DTLS handshake.

  Workaround: To support AP upgrades, configure a self-signed certificate for the controller as a workaround for the second use case. After updating all APs, delete the self-signed certificates. Then, revert to the SUDI certificate.

Note This workaround does not apply to Embedded Wireless Controllers running on Catalyst 9000 series switches. It is only valid for hardware appliance controllers such as Cisco Catalyst 9800-40, 9800-80, and 9800-L.

- The command **wireless management certificate ssc auth-token** is required for ensuring resilient operation of Cisco Catalyst 9800-CL controllers. It is mandatory to configure an explicit authentication token on all 9800-CL devices within the network. Additionally, this command is required when migrating

from an AireOS controller to a Cisco Catalyst 9800-CL controller to facilitate secure AP join, as the 9800-CL uses a self-signed certificate instead of a MIC used by AireOS controllers.

Prerequisites for controller self-signed certificate for wireless AP join

- Ensure that the VLAN interface is up and that the IP address is reachable.
- Ensure that the **ip http server** is enabled. For more information, see Enabling HTTP Server .
- Set the **clock calendar-valid** command appropriately. For more information, see *Configure clock calendar.*
- Check whether the PKI CA server is configured. If it is, delete the existing CA server configuration.

Note The **show crypto pki server** command output should not display anything.

Configure clock calendar (CLI)

Set the clock calendar to ensure accurate timekeeping on the device.

This configuration is necessary for time-sensitive operations and logging on the device.

Before you begin

Access to the device with appropriate privileges to enter global configuration mode.

Follow these steps to configure clock calendar:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable clock calendar.

Example:

```
Device(config)# clock calendar-valid
```

Step 3 Exit configuration mode.

Example:

```
Device(config)# exit
```

The clock calendar is now configured and operational on the device.

Enable HTTP server using CLI

Enable the HTTP server to allow web-based management of the device.

This configuration is applicable for devices that require HTTP access for management purposes.

Before you begin

Ensure you have administrative access to the device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enable the HTTP server on your IP or IPv6 system.

Example:

```
Device(config)# ip http server
```

Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. By default, the HTTP server uses the standard port 80.

Step 3 Enable the secure HTTP server on your IP or IPv6 system.

Example:

```
Device(config)# ip http secure-server
```

Enables the secure HTTP server on your IP or IPv6 system, providing encrypted access to the web interface.

Step 4 Exit configuration mode.

Example:

```
Device(config)# exit
```

The HTTP server is now enabled, allowing web-based management of the device.

Configure CA server (CLI)

Deploy a CA server to manage certificates for devices in the network.

This configuration is essential for environments that require secure communication through certificates.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Generate RSA keys for the CA server.

Example:

```
Device(config)# crypto key generate rsa general-keys modulus size_of_key_module label
keypair_name
```

Configures a certificate for the controller.

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.

Note
The recommended key-pair name is *WLC_CA* and key modulus is *2048* bits.

Step 3 Enable the IOS certificate server.

Example:

```
Device(config)# crypto pki server certificate_server_name
```

Enables IOS certificate server.

Note
The *certificate_server_name* must be the same name as the *keypair_name*.

Step 4 Configure the issuer name for the CA certificate.

Example:

```
Device(cs-server)# issuer-name O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC
```

Configures X.509 distinguished name for the issuer CA certificate.

Note
You need to configure the same *issuer-name* as suggested for AP join.

Step 5 Grant certificate requests automatically.

Example:

```
Device(cs-server)# grant auto
```

Grants certificate requests automatically.

Step 6 (Optional) Specify the hash function for the signature used in the granted certificates.

Example:

```
Device(cs-server)# hash sha256
```

Specifies the hash function for the signature used in the granted certificates.

Step 7 (Optional) Specify the lifetime in days of a CA certificate.

Example:

```
Device(cs-server)# lifetime ca-certificate time-interval
```

Specifies the lifetime in days of a CA certificate.

Step 8 (Optional) Specify the lifetime in days of a granted certificate.

Example:

```
Device(cs-server)# lifetime certificate time-interval
```

Specifies the lifetime in days of a granted certificate.

Step 9 Set the CA key and CA certificate archive format and password.

Example:

```
Device(cs-server)# database archive pkcs12 password 0 cisco123
```

Sets the CA key and CA certificate archive format and password to encrypt the file.

Step 10 Enable the certificate server.

Example:

```
Device(cs-server)# no shutdown
```

Enables the certificate server.

Note
Issue this command only after you have completely configured your certificate server.

Step 11 Return to privileged EXEC mode.

Example:

```
Device(cs-server)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The CA server is now configured and ready to manage certificates for devices in the network.

Configure trustpoint (CLI)

Configure a trustpoint to enable secure communication with a Certificate Authority (CA).

This configuration is necessary for devices that require secure communications and certificate management.

Before you begin

Ensure you have access to the device and the necessary permissions to configure trustpoints.

Follow these steps to configure trustpoint using CLI commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Generate RSA keys for the trustpoint.

Example:

```
Device(config)# crypto key generate rsa exportable general-keys modulus
size-of-the-key-modulus label label
```

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.

Step 3 Create a new trustpoint for an external CA server.

Example:

```
Device(config)# crypto pki trustpoint trustpoint_name
```

Here, *trustpoint_name* refers to the trustpoint name.

Note
Ensure that same names are used for key-pair (*label*) and *trustpoint_name*.

Step 4 Map RSA key with that of the trustpoint.

Example:

```
Device(ca-trustpoint)# rsakeypair RSA_key
```

Maps RSA key with that of the trustpoint.

- RSA_key—Refers to the RSA key pair label.
- key_size—Refers to the signature key length. The value ranges from 360 to 4096.

Step 5 Create subject name parameters for the trustpoint.

Example:

```
Device(ca-trustpoint)# subject-name subject_name
```

Specifies the subject name for the trustpoint.

Step 6 Check revocation.

Example:

```
Device(ca-trustpoint)# revocation-check none
```

Specifies whether to check for certificate revocation.

Step 7 Specify the hash algorithm.

Example:

```
Device(ca-trustpoint)# hash sha256
```

Specifies the hash algorithm to be used.

Step 8 Specify the serial number.

Example:

```
Device(ca-trustpoint)# serial-number
```

Specifies the serial number for the trustpoint.

Step 9 (Optional) Set certificate key-usage purpose.

Example:

```
Device(ca-trustpoint)# eku request server-auth client-auth
```

Sets the extended key usage for the certificate.

Step 10 Enable password for the trustpoint.

Example:

```
Device(ca-trustpoint)# password 0 password
```

Step 11 Enroll the URL for the CA server.

Example:

```
Device(ca-trustpoint)# enrollment url url
```

Replace the dummy IP with management VLAN interface IP of the controller where CA server is configured.

Step 12 Exit the configuration mode.

Example:

```
Device(ca-trustpoint)# end
```

The trustpoint is now configured and ready for use with the CA server.

Authenticate and enroll the PKI trustpoint with CA server

Authenticate and enroll the PKI trustpoint to establish a secure connection with the Certification Authority (CA) server.

This procedure is used in environments where a PKI trustpoint needs to be authenticated and enrolled with a Certification Authority (CA) server for certificate management.

Before you begin

Ensure that the Certification Authority (CA) server is reachable and the trustpoint is properly configured.

Follow these steps to authenticate and enroll the PKI trustpoint with CA server:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Authenticate the PKI trustpoint with the CA server.

Example:

```
Device(config)# crypto pki authenticate trustpoint-name
Certificate has the following attributes:
Fingerprint MD5: 64C5FC9A C581D827 C25FC3CF 1A7F42AC
Fingerprint SHA1: 6FAFF812 7C552783
6A8FB566 52D95849 CC2FC050
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
```

Fetches the CA certificate.

Step 3 Enroll the PKI trustpoint for a client certificate.

Example:

```
Device(config)# crypto pki enroll trustpoint-name
Enter following answers for UI interaction:
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
```

Step 4 Exit global configuration mode.

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI trustpoint is now authenticated and enrolled with the Certification Authority (CA) server, allowing for secure certificate management.

Tag wireless management trustpoint name

Tag a wireless management trustpoint name for identification and management purposes.

This configuration is used in environments where wireless management trustpoints need to be clearly identified for management and security.

Before you begin

Ensure you have the necessary privileges to enter global configuration mode.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Tag the wireless management trustpoint name.

Example:

```
Device(config)# wireless management trustpoint trustpoint-name
```

Step 3 Exit global configuration mode.

Example:

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The wireless management trustpoint name is successfully tagged and can be used for management purposes.

Disable PKI server

Disable the PKI server to prevent it from issuing certificates.

This procedure is used in scenarios where the PKI server is no longer needed or must be temporarily disabled.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a new server for an external CA server.

Example:

```
Device(config)# crypto pki server certificate-server-name
```

The *certificate-server-name* refers to the server name.

Note
Here, server_name refers to the server name.

Step 3 Disable the certificate server.

Example:

```
Device(cs-server)# shutdown
```

Update purpose string to \

Step 4 Return to privileged EXEC mode.

Example:

```
Device(cs-server)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI server is now disabled and will not issue certificates.

Verify controller certificates for wireless AP join

Use these commands to verify controller certificates for wireless access point (AP) join.

To view the CA server details, use the command:

```
Device# show crypto pki server
Certificate Server WLC_CA:
Status: enabled
State: enabled
Server's configuration is locked (enter "shut" to unlock it)
Issuer name: O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC
CA cert fingerprint: 79A3DBD5 59A7E384 73ABD152 C133F4E2
Granting mode is: auto
Last certificate issued serial number (hex): 1
CA certificate expiration timer: 12:04:00 UTC Mar 8 2029
CRL NextUpdate timer: 18:04:00 UTC Mar 11 2019
Current primary storage dir: nvram:
Database Level: Minimum - no cert data written to storage
```

To view the trustpoint details, use this command:

```
Device# show crypto pki trustpoint ewlc-tp1 status
Trustpoint ewlc-tp1:
...
State:
Keys generated ..... Yes (General Purpose, exportable)
Issuing CA authenticated ..... Yes
Certificate request(s) ..... Yes
```

To view the wireless management trustpoint details, use this command:

```
Device# do show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : 4a5d777c5b2071c17faef376febc08398702184e
Private key Info : Available
FIPS suitability : Not Applicable
```

To view the Hypertext Transfer Protocol (HTTP) server status, use this command:

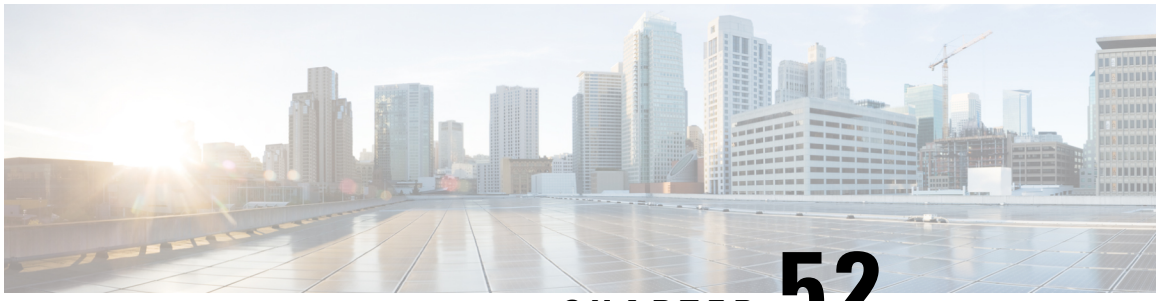
```
Device# show ip http server status | include server status
HTTP server status: Enabled
HTTP secure server status: Enabled
```



PART VII

Quality of Service

- Quality of Service, on page 545
- Wireless Auto-QoS, on page 581
- Native Profiling, on page 587



CHAPTER 52

Quality of Service

- Wireless Quality of Service, on page 545
- Wireless QoS targets, on page 546
- Precious metal policies for wireless QoS, on page 547
- Prerequisites for wireless QoS, on page 548
- Restrictions for QoS on wireless targets, on page 548
- Metal Policy Format, on page 550
- How to apply Bi-Directional Rate Limiting, on page 560
- How to apply Per Client Bi-Directional Rate Limiting, on page 567
- How to Configure Wireless QoS, on page 571
- Configure custom QoS mapping (CLI), on page 576
- Configure a DSCP-to-user priority mapping exception, on page 577
- Configure trust upstream DSCP value, on page 579

Wireless Quality of Service

A wireless Quality of Service policy is a network management policy that

- prioritizes specific types of wireless traffic by giving preferential treatment
- applies different rules to SSID and client targets in both upstream and downstream directions, and
- supports traffic marking, rate limiting (policing), mobility features, and compatibility with advanced controller functions.

- Upstream traffic: The flow of data from a wireless source to a wired target.
- Downstream traffic: The flow of data from a wired source to a wireless target.
- Target: The entity (SSID or client) where the QoS policy is enforced.

Additional reference information

- Without QoS, network devices transmit packets with best-effort service, offering no guarantees for reliability, delay bounds, or throughput. Wireless QoS policies enhance the network by ensuring that traffic with higher priority receives preferential treatment. This policy improves overall performance for critical applications.

- Applying a wireless QoS policy to prioritize voice traffic on an SSID ensures that calls suffer less latency or jitter compared to general web browsing traffic.
- Rate limiting is used to prevent a single client from consuming excessive bandwidth and to maintain fair usage for all wireless clients.
- A network with no configured QoS policies treats all wireless traffic equally, which can lead to poor performance for delay-sensitive applications.
- QoS policies designed for wired networks do not automatically apply to wireless environments, as wireless traffic has unique constraints.

Wireless QoS targets

This section describes the various wireless QoS targets available on a device.

Service set identifiers policies

An Service Set Identifiers (SSID) policy is a wireless network configuration policy that

- controls the application of QoS settings to a wireless SSID in both ingress and egress directions
- applies per AP and per SSID, and
- allows configuration of policing and marking actions on SSID traffic.

If an SSID policy is not configured, no QoS policy is applied to the SSID.

Client policies

Client policies are applicable in the ingress and egress direction. You can configure policing and marking policies on clients. AAA override is also supported.

Supported QoS features on wireless targets

Wireless controllers support various QoS features to manage traffic and ensure optimal performance for both SSIDs and client devices. These tables show the supported features, applicable directions, and configuration modes for wireless targets.

This table describes the various features available on wireless targets.

Table 28: QoS features available on wireless targets

Target	Features	Direction where policies Are applicable
SSID	• Set • Police • Drop	Upstream and downstream

Target	Features	Direction where policies Are applicable
Client	• Set • Police • Drop	Upstream and downstream

Note For Drop support, the drop action is achieved by this configuration:

```
police <rate>
   conform-action drop
   exceed-action drop
```

Direct **action drop** is not supported.

Precious metal policies for wireless QoS

Precious metal policies for wireless QoS are system-defined QoS policies

- assign different service levels to wireless network traffic based on pre-set categories,
- remain unmodifiable, cannot be removed by administrators, and
- affect packet attributes, such as 802.11e (WMM) and DSCP fields, when policies are applied.
- Platinum: Used for VoIP clients, assigns the highest priority.
- Gold: Used for video clients, assigns high priority, lower than platinum.
- Silver: Used for best-effort traffic, assigns standard priority.
- Bronze: Used for Non-Real-Time (NRT) traffic, assigns the lowest priority.

Additional reference information

Preconfigured precious metal policies are available on wireless controllers. Administrators cannot modify or delete these policies. AAA mechanisms may push client metal policies. These policies determine packet scheduling and marking on the network.

In FlexConnect local switching mode, APs do not enforce QoS metal policy ceiling limits for upstream traffic. Limit enforcement occurs at the controller exit point. The AP does not change DSCP values.

See the Metal policy format, on page 550 section for more information about metal policy formats.

See the Metal policy map section for more information about metal policies.

Precious metal policies for wireless QoS

- Assigning the platinum policy to VoIP devices ensures minimal latency for voice communication.

- Assigning the gold policy to video streaming clients optimizes traffic for media applications.
- The silver policy applied to laptops browsing the internet handles general best-effort data.
- The bronze policy used for devices performing background updates minimizes their network priority.

- Custom user-defined QoS policies that administrators can modify or remove are not considered precious metal policies.
- Policies that do not map traffic types to platinum, gold, silver, or bronze levels are not included.

Prerequisites for wireless QoS

Before you configure wireless QoS, make sure you understand key concepts and network factors to ensure effective deployment.

The required prerequisites include:

- Understand wireless concepts and network topologies.
- Understand QoS implementation.
- Modular QoS CLI (MQC). To learn more about Modular QoS CLI, see the MQC guide.
- Understand the types of applications used and the traffic patterns on your network.
- Understand your network's bandwidth requirements and speed.

Restrictions for QoS on wireless targets

QoS policy application on wireless targets (such as SSIDs, BSSIDs, and wireless clients) has these key restrictions and considerations.

General Restrictions

- QoS policy application on wireless targets (such as Service Set Identifier (SSID), BSSID, and wireless clients) includes these key restrictions and considerations:

Hierarchical (parent policy and child policy) QoS is not supported.

- You can assign only one policy per target for each direction.
- You can use only BSSID and client targets for both directions.
- These policy formats are supported:
  - QoS Policy Action
    - Police:

      ```
      police [cir | rate] bps [conform-action action] [exceed-action action]
      ```

      You can set the policer action type to **transmit** or **drop**.

- Set:

```
set dscp
set wlan user-priority
```



Note **set wlan user-priority** (downstream only; BSSID only)

- QoS Policy Classification

```
match [not] access-group
        match [not] dscp
        match [not] protocol
```

AP Side Restrictions

- In Cisco Embedded Wireless Controller, FlexConnect local switching, and Software-Defined Access (SDA) deployments, the AP enforces QoS policies and rate-limiting actions at the per-flow (5-tuple) level, not per client.
- For FlexConnect local switching with local authentication and AAA override enabled (using an external AAA server), you can use only air space VLAN and ACL as AAA overrides. QoS and other overrides are not available.
- Rate limiting per SSID is not supported in FlexConnect local switching mode. For rate limiting to work as expected, FlexConnect central switching mode should be used.

Control Plane Rate Limiting and Policing

You do not need to configure control plane rate limiting or policing. Built-in mechanisms, such as policers, protect the CPU from control plane traffic. Migrations from AireOS to IOS-XE handle this automatically.

Restriction: Client connectivity and TCP packet drops on SSIDs with AAA rate limiting

Do not configure rate limits or AAA overrides on Cisco ISE for any SSID where AAA rate limiting is applied. These settings can cause the APs in local mode to drop TCP packets. In such a scenario, use Modular QoS CLI (MCQ)-based method only for configuring Quality of Service (QoS) and rate limiting.

Behavior explanation

When a client connects to an SSID with AAA rate limiting configured on APs in local mode, the AP forwards ICMP, DHCP, and DNS packets to the controller but drops TCP packets. This issue is caused when rate limits are configured on Cisco ISE and AAA overrides are in place. You can verify that TCP packets are dropped using the **show datapath drop trace** comand.

Workaround

Review the configuration of rate limits and consider removing them or adjusting the settings to avoid dropping TCP packets. Use Modular QoS CLI (MCQ)-based method only for configuring Quality of Service (QoS)

and rate limiting. For more information about the MCQ-based method, see Configure QoS Rate Limit on Catalyst 9800 Wireless Controllers.

Verify client connectivity

- To observe the AP configuration, use this command.

```
show ap dot11 5ghz network
```

- To verify the AP configuration, use this command.

```
show ap dot11 5ghz network
```

- To check the client summary, use this command.

```
show wireless client summary
```

- To verify client details and troubleshoot connectivity issues, use this command:

```
show wireless client mac-address MAC-address detail
```

Metal Policy Format

Metal policy format

Metal Policies are system defined, and you cannot change it or delete it.

Metal policies are system defined. You cannot change or delete them. There are four levels of metal policy - Platinum, Gold, Silver, and Bronze.



Note Each metal policy defines a DSCP ceiling so that the DSCP or the UP marking does not exceed a certain value.

For Platinum the value is 46, Gold is AF41, Silver is 22, and Bronze is CS1.

Table 29: Metal Policy Map Formats

Policy Name	Policy-map Format	Class-map Format
platinum	policy-map platinum class cm-dscp-34 set dscp af41 class cm-dscp-45 set dscp 45 class cm-dscp-46 set dscp ef class cm-dscp-47 set dscp 47	class-map match-any cm-dscp-34 match dscp af41 class-map match-any cm-dscp-45 match dscp 45 class-map match-any cm-dscp-46 match dscp ef class-map match-any cm-dscp-47 match dscp 47 class-map match-any cm-dscp-0 match dscp default
gold	policy-map gold class cm-dscp-45 set dscp af41 class cm-dscp-46 set dscp af41 class cm-dscp-47 set dscp af41	
silver	policy-map silver class cm-dscp-34 set dscp default class cm-dscp-45 set dscp default class cm-dscp-46 set dscp default class cm-dscp-47 set dscp default	
bronze	policy-map bronze class cm-dscp-0 set dscp cs1 class cm-dscp-34 set dscp cs1 class cm-dscp-45 set dscp cs1 class cm-dscp-46 set dscp cs1 class cm-dscp-47 set dscp cs1	

Table 30: Metal Policy Map Formats (UP)

Policy Name	Policy-map Format	Class-map Format
platinum-up	policy-map platinum-up class cm-dscp-set1-for-up-4 set dscp af41 class cm-dscp-set2-for-up-4 set dscp af41 class cm-dscp-for-up-5 set dscp af41 class cm-dscp-for-up-6 set dscp ef class cm-dscp-for-up-7 set dscp ef	class-map match-any cm-dscp-for-up-0 match dscp default match dscp cs2 class-map match-any cm-dscp-for-up-1 match dscp cs1 class-map match-any cm-dscp-set1-for-up-4 match dscp cs3 match dscp af31 match dscp af32 match dscp af33 class-map match-any cm-dscp-set2-for-up-4 match dscp af41 match dscp af42 match dscp af43 class-map match-any cm-dscp-for-up-5 match dscp cs4 match dscp cs5 class-map match-any cm-dscp-for-up-6 match dscp 44 match dscp ef class-map match-any cm-dscp-for-up-7 match dscp cs6 match dscp cs7
gold-up	policy-map gold-up class cm-dscp-for-up-6 set dscp af41 class cm-dscp-for-up-7 set dscp af41	
silver-up	policy-map silver-up class cm-dscp-set1-for-up-4 set dscp default class cm-dscp-set2-for-up-4 set dscp default class cm-dscp-for-up-5 set dscp default class cm-dscp-for-up-6 set dscp default class cm-dscp-for-up-7 set dscp default	
bronze-up	policy-map bronze-up class cm-dscp-for-up-0 set dscp cs1 class cm-dscp-for-up-1 set dscp cs1 class cm-dscp-set1-for-up-4 set dscp cs1 class cm-dscp-set2-for-up-4 set dscp cs1 class cm-dscp-for-up-5 set dscp cs1 class cm-dscp-for-up-6 set dscp cs1 class cm-dscp-for-up-7 set dscp cs1	

Table 31: Metal Policy Map Formats (CLWMM/CLNON)

Policy Name	Policy-map Format	Class-map Format
clwmm-platinum	policy-map clwmm-platinum class voice-plat set dscp ef class video-plat set dscp af41 class class-default set dscp default	class-map match-any voice-plat match dscp ef class-map match-any video-plat match dscp af41 class-map match-any voice-gold match dscp ef class-map match-any video-gold match dscp af41
clwmm-gold	policy-map clwmm-gold class voice-gold set dscp af41 class video-gold set dscp af41 class class-default set dscp default	
clnon-wmm-platinum	policy-map clnon-wmm-platinum class class-default set dscp ef	
clnon-wmm-gold	policy-map clnon-wmm-gold class class-default set dscp af41	
clsilver	policy-map clsilver class class-default set dscp default	
clbronze	policy-map clbronze class class-default set dscp cs1	

Auto QoS policy format

This section provides the Auto QoS policy format, including policy-map and class-map configurations.

Policy name	Policy-map format	Class-map format
enterprise-avc		

Policy name	Policy-map format	Class-map format
	policy-map AutoQos-4.0-wlan-ET-SSID-Input-AVC-Policy class AutoQos-4.0-wlan-Voip-Data-Class set dscp ef class AutoQos-4.0-wlan-Voip-Signal-Class set dscp cs3 class AutoQos-4.0-wlan-Multimedia-Conf-Class set dscp af41 class AutoQos-4.0-wlan-Transaction-Class set dscp af21 class AutoQos-4.0-wlan-Bulk-Data-Class set dscp af11 class AutoQos-4.0-wlan-Scavanger-Class set dscp cs1 class class-default set dscp default policy-map AutoQos-4.0-wlan-ET-SSID-Output-Policy class AutoQos-4.0-RT1-Class set dscp ef	class-map match-any AutoQos-4.0-wlan-Voip-Data-Class match dscp ef class-map match-any AutoQos-4.0-wlan-Voip-Signal-Class match protocol skinny match protocol cisco-jabber-control match protocol sip match protocol sip-tls class-map match-any AutoQos-4.0-wlan-Multimedia-Conf-Class match protocol cisco-phone-video match protocol cisco-jabber-video match protocol ms-lync-video match protocol webex-media class-map match-any AutoQos-4.0-wlan-Transaction-Class match protocol cisco-jabber-im match protocol ms-office-web-apps

Policy name	Policy-map format	Class-map format
	class AutoQos-4.0-RT2-Class set dscp af31 class class-default	match protocol salesforce match protocol sap class-map match-any AutoQos-4.0-wlan-Bulk-Data-Class match protocol ftp match protocol ftp-data match protocol ftps-data match protocol cifs class-map match-any AutoQos-4.0-wlan-Scavanger-Class match protocol netflix match protocol youtube match protocol skype match protocol bittorrent class-map match-any AutoQos-4.0-RT1-Class match dscp ef match dscp cs6

Policy name	Policy-map format	Class-map format
		class-map match-any AutoQos-4.0-RT2-Class match dscp cs4 match dscp cs3 match dscp af41
voice	policy-map platinum-up class dscp-for-up-4 set dscp 34 class dscp-for-up-5 set dscp 34 class dscp-for-up-6 set dscp 46 class dscp-for-up-7 set dscp 46 policy-map platinum class cm-dscp-34 set dscp 34 class cm-dscp-46 set dscp 46	-

Policy name	Policy-map format	Class-map format
guest	Policy Map AutoQos-4.0-wlan-GT-SSID-Output-Policy Class class-default set dscp default Policy Map AutoQos-4.0-wlan-GT-SSID-Input-Policy Class class-default set dscp default	-
port (only applies to Local Mode)	policy-map AutoQos-4.0-wlan-Port-Output-Policy class AutoQos-4.0-Output-CAPWAP-C-Class priority level 1 class AutoQos-4.0-Output-Voice-Class priority level 2 class class-default ip access-list extended AutoQos-4.0-Output-Acl-CAPWAP-C permit udp any eq 5246 16666 any	class-map match-any AutoQos-4.0-Output-CAPWAP-C-Class match access-group name AutoQos-4.0-Output-Acl-CAPWAP-C class-map match-any AutoQos-4.0-Output-Voice-Class match dscp ef

Architecture for voice, video and integrated data (AVVID)

The table lists how AVVID service classes map to the IETF DiffServ DSCP values and IEEE 802.11e categories. Use this information to classify traffic and assign appropriate QoS markings for voice, video, and integrated data services.

Table 32: Mapping of IETF diffServ, DSCP, and IEEE 802.11e

IETF diffServ service class	DSCP	IEEE 802.11e	
		User priority	Access category
Network Control	(CS7) CS6	0	AC_BE
Telephony	EF	6	AC_VO
VOICE-ADMIT	44	6	AC_VO
Signaling	CS5	5	AC_VI
Multimedia Conferencing	AF41 AF42 AF43	4	AC_VI
Real-Time Interactive	CS4	5	AC_VI
Multimedia Streaming	AF31 AF32 AF33	4	AC_VI
Broadcast Video	CS3	4	AC_VI
Low-Latency Data	AF21 AF22 AF23	3	AC_BE
OAM	CS2	0	AC_BE
High-Throughput Data	AF11 AF12 AF13	2	AC_BK
Standard	DF	0	AC_BE
Low-Priority Data	CS1	1	AC_BK
Remaining	Remaining	0	-

How to apply Bi-Directional Rate Limiting

Bidirectional rate limiting

A bidirectional rate limit is a wireless network traffic management feature that

- establishes configurable rate limits for both upstream and downstream traffic directions
- enables administrators to set individual limits per direction directly on the WLAN, overriding QoS profile values, and global controller configurations, and
- supports prioritization of client groups by assigning them to specific QoS profiles.

QoS profiles

There are four distinct QoS profiles to configure rate limits:

- Gold
- Platinum
- Silver
- Bronze

Additional reference information

- Apply rate limiting directly to a WLAN. This action overrides both global QoS settings and QoS profiles for the controller and clients.
- Bidirectional rate limits apply to all clients associated with a given SSID. Every client connected to the same SSID has identical rate restrictions.
- Set throughput limits to control wireless client performance for both traffic directions. Assign service priority to specific client sets for prioritization.

Configuration guidance

- Select a QoS profile and configure the rate limiting parameters for upstream and downstream directions. Setting a parameter to `0` disables rate limiting for that direction.
- Assign a QoS profile to each WLAN to determine the rate limits for all connected clients.

Scenario considerations

- Configure bidirectional rate limits on both the Anchor and Foreign controllers in mobility Anchor–Foreign controller setups. Use identical configuration across both controllers to prevent feature failure.
- The feature is supported in guest anchor scenarios, including IRCM guest deployments where AireOS devices function as guest anchor or guest foreign.
- Cisco Catalyst 9800 Series Wireless Controller uses a policing option to enforce bidirectional rate limits.

Bidirectional rate limiting

If a guest SSID uses the Bronze QoS profile and sets both upstream and downstream limits to specific values, all guests on that SSID have consistent rate restrictions, regardless of any global QoS or profile settings.

Requirements for bidirectional rate limiting configuration

Ensure you meet the essential requirements before configuring bidirectional rate limiting.

To configure bidirectional rate limiting, ensure these prerequisites are met:

- Apply the client metal policy through AAA override.
- Specify the metal policy on the Identity Services Engine (ISE) server.
- Enable AAA override on the policy profile.

Configure the metal policy on the SSID

Apply a metal service policy to a wireless SSID using a WLAN policy profile.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a WLAN policy profile and enter wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy policy-profile-name
```

Step 3 Add a user-defined description to the new wireless policy.

Example:

```
Device(config-wireless-policy)# description description
```

Step 4 Set the platinum policy for input.

Example:

```
Device(config-wireless-policy)# service-policy input input-policy
```

Step 5 Set the platinum policy for output.

Example:

```
Device(config-wireless-policy)# service-policy output output-policy
```

The system applies the configured policy profile to the SSID. It enforces platinum-level input and output service policies.

Configure a metal policy on your client device

Set up a wireless metal policy profile that uses AAA override on your client device.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile and enter wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy policy-profile-name
```

Step 3 Add a user-defined description to the new wireless policy.

Example:

```
Device(config-wireless-policy)# description description
```

Step 4 Enable AAA override on your WLAN.

Example:

```
Device(config-wireless-policy)# aaa-override
```

Note
After you enable AAA override and the ISE server starts sending policy, the client policy defined in the service-policy client does not take effect.

With AAA override enabled, your wireless policy profile allows external RADIUS authentication and policy control.

Configure bidirectional rate limiting for all traffic

Enforce bidirectional rate limiting on all traffic by using policy maps.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a named object to apply policies to traffic classes.

Example:

```
Device(config)# policy-map policy-map
```

Policy-map names can contain alphabetic, hyphen, or underscore characters. The names are case sensitive. Names can be up to 40 characters long.

Step 3 Associate a class map with the policy map and enter policy-map class configuration mode.

Example:

```
Device(config-pmap)# class class-map-name
```

Step 4 Configure traffic policing for your traffic class.

Example:

```
Device(config-pmap-c)# police rate
```

Valid values are 8,000 to 200,000,000 bps.

The bidirectional rate limiting policy limits all traffic that matches the class map.

Configure Bi-Directional Rate Limiting Based on Traffic Classification

Procedure

Step 1 **configure terminal**

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 **policy-map** *policy-map*

Example:

```
Device(config)# policy-map policy-sample2
```

Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.

Step 3 **class** *class-map-name*

Example:

```
Device(config-pmap)# class class-sample-youtube
```

Associates a class map with the policy map, and enters policy-map class configuration mode.

Step 4 **police** *rate*

Example:

```
Device(config-pmap-c)# police 1000000
```

Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

Step 5 **conform-action drop**

Example:

```
Device(config-pmap-c-police)# conform-action drop
```

Specifies the drop action to take on packets that conform to the rate limit.

Step 6 **exceed-action drop**

Example:

```
Device(config-pmap-c-police)# exceed-action drop
```

Specifies the drop action to take on packets that exceeds the rate limit.

Step 7 **exit**

Example:

```
Device(config-pmap-c-police)# exit
```

Exits the policy-map class configuration mode.

Step 8 **set dscp default**

Example:

```
Device(config-pmap-c)# set dscp default
```

Sets the DSCP value to default.

Step 9 **police** *rate*

Example:

```
Device(config-pmap-c)# police 500000
```

Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

Step 10 **exit**

Example:

```
Device(config-pmap-c)# exit
```

Exits the policy-map class configuration mode.

Step 11 **exit**

Example:

```
Device(config-pmap)# exit
```

Exits the policy-map configuration mode.

Step 12 **class-map match-any** *class-map-name*

Example:

```
Device(config)# class-map match-any class-sample-youtube
```

Selects a class map.

Step 13 **match protocol** *protocol*

Example:

```
Device(config-cmap)# match protocol youtube
```

Configures the match criteria for a class map on the basis of the specified protocol.

Apply bidirectional rate limiting policy map to policy profile

Apply bidirectional rate limiting to your wireless network by attaching a policy map to a policy profile.

Procedure

Step 1 Enter global configuration mode to begin setup.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile to enter wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy policy-profile-name
```

Step 3 Add a user-defined description to your new wireless policy.

Example:

```
Device(config-wireless-policy)# description description
```

Step 4 Set the input client service policy to platinum.

Example:

```
Device(config-wireless-policy)# service-policy client input platinum-up
```

Step 5 Set the output client service policy to platinum.

Example:

```
Device(config-wireless-policy)# service-policy client output output-policy
```

Step 6 Set the input service policy to platinum.

Example:

```
Device(config-wireless-policy)# service-policy input input-policy
```

Step 7 Set the output service policy to platinum.

Example:

```
Device(config-wireless-policy)# service-policy output platinum
```

Your wireless policy profile applies bidirectional rate limiting based on the service policies you set.

Apply metal policy with bidirectional rate limiting

Use a metal policy to enforce bidirectional bandwidth limits on wireless traffic.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure WLAN policy profile and enter wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy policy-profile-name
```

Step 3 Add a description for the new wireless policy.

Example:

```
Device(config-wireless-policy)# description description
```

Step 4 Assign 'platinum' as the input client service policy.

Example:

```
Device(config-wireless-policy)# service-policy client input input-policy
```

Step 5 Assign 'platinum' as the output client service policy.

Example:

```
Device(config-wireless-policy)# service-policy client output output-policy
```

Step 6 Assign 'platinum' as the input service policy.

Example:

```
Device(config-wireless-policy)# service-policy input input-policy
```

Step 7 Assign 'platinum' as the output service policy.

Example:

```
Device(config-wireless-policy)# service-policy output platinum
```

Step 8 Exit the policy configuration mode.

Example:

```
Device(config-wireless-policy)# exit
```

Step 9 Create a named object to apply policies to traffic classes.

Example:

```
Device(config)# policy-map policy-sample 1
```

Policy map names can contain alphabetic characters, hyphens, or underscores. They are case sensitive and can be up to 40 characters long.

Step 10 Associate a class map with the policy map, and enter configuration mode for the specified system class.

Example:

```
Device(config-pmap)# class class-map-name
```

Step 11 Configure traffic policing.

Example:

```
Device(config-pmap-c)# police 500,000
```

Valid values range from 8,000 to 200,000,000 bytes.

The metal policy enforces bidirectional rate limiting so wireless traffic meets the input and output bandwidth limits you specified.

How to apply Per Client Bi-Directional Rate Limiting

Per-client bidirectional rate limiting

A per-client bidirectional rate limit is a wireless traffic control feature that

- enforces, for each wireless client, bandwidth caps separately on both upstream and downstream traffic,
- ensures that each client receives a single aggregate limit, no matter how many concurrent flows or streams are active,
- addresses limitations of legacy per-flow rate limiting on 802.11ac Wave 2 APs in Flex local switching mode.

Additional reference information

Previously, per-flow rate limiting capped each client stream, such as a YouTube stream or an FTP transfer, independently. As a result, clients could exceed the intended per-client bandwidth limit.

With per-client bidirectional rate limits, the total bandwidth used by all of a client's streams cannot exceed the configured limit. This restriction applies no matter how many streams are active.

Per-client bidirectional rate limiting

If a controller limits each client to 1000 Kbps (1 megabit per second), and a client initiates both a YouTube and an FTP stream, both streams together will share the 1000 Kbps (1 megabit per second) limit, ensuring the client cannot exceed the cap.

Prerequisites for per-client bidirectional rate limiting

- You can use this feature only with a QoS client policy. Ensure the policy profile contains only a QoS Policy or a policy target as client.
- If the policy map includes class default with a valid police rate value, the access point applies this rate limit to the the data traffic flow for the client.

Restrictions on per-client bidirectional rate limiting

- If the policy map has a class map other than the Default class map, the per-client rate limit does not work on the AP.

Configure per-client bidirectional rate limiting (GUI)

Set limits on upload and download data rates for each client. This ensures fair bandwidth distribution.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click the Policy Profile Name.

The **Edit Policy Profile** window is displayed.

Note
The **Edit Policy Profile** window is displayed and configured in default class map only.

Step 3 Choose the **QOS And AVC** tab.

Step 4 In the **QoS Client Policy** settings, choose the policies from the **Egress** and **Ingress** drop-down lists.

Note
You need to apply the default policy map to the QoS Client Policy.

Step 5 Click **Update & Apply to Device**.

The selected policies in the policy profile enforce bidirectional rate limiting for each client.

Verify per client bi-directional rate limiting

To verify whether per client is applied in AP, use this command:

```
Device# show rate-limit client
Config:
            mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in nrt_rate_out nrt_rate_in
 nrt_burst_out nrt_burst_in
A0:D3:7A:12:6C:5E   0           0          0            0           0            0
   0             0            0
Statistics:
            name     up down
        Unshaped      0    0
  Client RT pass 697610 8200
 Client NRT pass      0    0
 Client RT drops      0    0
Client NRT drops      0   16
               9    180    0
Per client rate limit:
             mac vap rate_out rate_in            policy
A0:D3:7A:12:6C:5E   0       88      23 per_client_rate_2
```

Configure bi-directional rate limiting using AAA override (CLI)

Enable WLAN policy profiles to enforce upstream and downstream bandwidth limits based on RADIUS server (e.g., Cisco ISE) attributes through AAA override.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile and enter wireless policy configuration mode.

Example:

```
Device (config)# wireless profile policy profile-name
```

Step 3 Configure AAA override to apply policies coming from the AAA server or Cisco Identity Services Engine (ISE) server.

Example:

```
Device(config-wireless-policy)# aaa-override
```

These attributes are available in the RADIUS server:

- Airespace-Data-Bandwidth-Average-Contract: 8001
- Airespace-Real-Time-Bandwidth-Average-Contract: 8002
- Airespace-Data-Bandwidth-Burst-Contract: 8003
- Airespace-Real-Time-Bandwidth-Burst-Contract: 8004
- Airespace-Data-Bandwidth-Average-Contract-Upstream: 8005
- Airespace-Real-Time-Bandwidth-Average-Contract-Upstream: 8006
- Airespace-Data-Bandwidth-Burst-Contract-Upstream: 8007
- Airespace-Real-Time-Bandwidth-Burst-Contract-Upstream: 8008

Note
8001, 8002, 8003, 8004, 8005, 8006, 8007, and 8008 are example rate-limit values.

The WLAN policy profile is configured to honor bi-directional rate limiting instructions received via AAA override from the RADIUS server.

Verify bi-directional rate-limit

To verify the bi-directional rate limit, use this command:

```
Device# show wireless client mac-address E8-8E-00-00-00-71 detailClient MAC Address :
e88e.0000.0071
```

```
Client MAC Type     : Universally Administered Address
Client IPv4 Address : 192.0.2.1
Client Username     : e88e00000071
AP MAC Address      : 0a0b.0c00.0200
AP Name             : AP6B8B4567-0002
AP slot             : 0
Client State        : Associated
Policy Profile      : dnas_qos_profile_policy
Flex Profile        : N/A
Wireless LAN Id     : 10
WLAN Profile Name   : QoS_wlan
Wireless LAN Network Name (SSID): QoS_wlan
BSSID : 0a0b.0c00.0200
Connected For       : 28 seconds
Protocol            : 802.11n - 2.4 GHz
Channel             : 1
Client IIF-ID       : 0xa0000034
Association Id      : 10
Authentication Algorithm : Open System
Idle state timeout  : N/A
Session Timeout     : 1800 sec (Remaining time: 1777 sec)
Session Warning Time : Timer not running
Input Policy Name    : None
Input Policy State   : None
Input Policy Source  : None
Output Policy Name   : None
Output Policy State  : None
Output Policy Source : None
WMM Support          : Enabled
U-APSD Support       : Disabled
Fastlane Support     : Disabled
Client Active State  : In-Active
Power Save           : OFF
Supported Rates : 1.0,2.0,5.5,6.0,9.0,11.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 8005 (kbps)
  QoS Realtime Average Data Rate Upstream    : 8006 (kbps)
  QoS Burst Data Rate Upstream               : 8007 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 8008 (kbps)
  QoS Average Data Rate Downstream           : 8001 (kbps)
  QoS Realtime Average Data Rate Downstream  : 8002 (kbps)
  QoS Burst Data Rate Downstream             : 80300 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 8004 (kbps)
```

To verify the rate-limit details from the AP terminal, use this command

```
Device# show rate-limit client
Config:
mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in nrt_rate_out nrt_rate_in nrt_burst_out
 nrt_burst_in
00:1C:F1:09:85:E7 0 8001 8002 8003 8004 8005 8006 8007 8008
Statistics:
name up down
Unshaped 0 0
Client RT pass 0 0
Client NRT pass 0 0
Client RT drops 0 0
Client NRT drops 0 0
Per client rate limit:
mac vap rate_out rate_in policy
```

How to Configure Wireless QoS

Configure a policy map with class map (GUI)

Define and apply a QoS policy map with associated class maps to control network traffic behavior.

Use this task when you need to create or update a QoS policy map containing class maps, specifying how different types of network traffic are marked, policed, or dropped.

Before you begin

Use these steps to configure a policy map with class map using the GUI.

Procedure

Step 1 Choose **Configuration** > **Services** > **QoS**.

Step 2 Click **Add** to view the **Add QoS** window.

Step 3 In the text box next to **Policy Name**, enter the name of the new policy map.

Step 4 Click **Add Class-Maps**.

Step 5 Configure **AVC** based policies or **User Defined** policies. To enable **AVC** based policies, and configure the following:

a) Choose either **Match Any** or **Match All**.

b) Choose the required **Mark Type**. If you choose **DSCP** or **User Priority**, you must specify the appropriate **Mark Value**.

c) Check the **Drop** check box to drop traffic from specific sources.

Note
When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

d) Select the required protocols from the **Available Protocol(s)** list based on the chosen **Match Type**. Move them to the **Selected Protocol(s)** list. The system drops traffic from these selected protocols.

e) Click **Save**.

Note
To add more Class Maps, repeat steps 4 and 5.

Note
To add more Class Maps, repeat steps 4 and 5.

Step 6 To enable the **User-Defined** QoS policy, configure these options:

a) Choose either **Match Any** or **Match All**.

b) From the drop-down list, choose either **ACL** or **DSCP** as the **Match Type**. Specify the appropriate **Match Value**.

c) Choose the required **Mark Type** to associate with the mark label. If you choose *DSCP*, you must specify an appropriate **Mark Value**.

d) Check the **Drop** check box to drop traffic from specific sources.

Note
When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

e) Click **Save**.

Note
To define actions for all the remaining traffic, in the Class Default, choose either **Mark**, **Police(kbps)**, or both as appropriate.

Step 7 Click **Save & Apply to Device**.

The system deploys the defined policy map with its class maps and associated actions. QoS settings are enforced on network traffic as specified.

Configure a class map (CLI)

Define and customize a class map to identify and match specific network traffic, such as voice and video, using CLI commands.

Use this procedure to configure class maps for voice and video traffic:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Create a class map.

Example:

```
Device(config)# class-map class-map-name
```

Step 3 Match the DSCP value in IPv4 and IPv6 packets.

Example:

```
Device(config-cmap)# match dscp dscp-value
```

Note
By default, the class map uses match-all.

Step 4 Exit class map configuration mode and return to privileged EXEC mode.

Example:

```
Device(config-cmap)# end
```

Step 5 Verify class map details.

Example:

```
Device# show class-map class_map_name
```

You have configured the class map with the specified matching criteria. It is ready for use in traffic policies.

Configuring Policy Profile to Apply QoS Policy (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click the name of the policy profile.

Step 3 In the **Edit Policy Profile** window, click the **QoS and AVC** tab.

Step 4 Under **QoS SSID Policy**, choose the appropriate **Ingress** and **Egress** policies for WLANs.

Note
The ingress policies can be differentiated from the egress policies by the suffix *-up*. For example, the Platinum ingress policy is named *platinum-up*.

Step 5 Under **QoS Client Policy**, choose the appropriate **Ingress** and **Egress** policies for clients.

Step 6 Click **Update & Apply to Device**.

Note
Only custom policies are displayed under **QoS Client Policy**. AutoQoS policies are auto generated and not displayed for user selection.

Configure policy profile to apply QoS policy (CLI)

Apply a QoS policy to a WLAN policy profile using CLI.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the WLAN policy profile. Enter the wireless policy configuration mode.

Example:

```
Device(config)# wireless profile policy profile-policy
```

Step 3 Apply the policy.

Example:

```
Device(config-wireless-policy)# service-policy client input policy-map-client
```

These options are available:

- **input**—Assigns the client policy for ingress direction on the policy profile.
- **output**—Assigns the client policy for egress direction on the policy profile.

Step 4 Apply the policy to the Basic Service Set Identifier (BSSID).

Example:

```
Device(config-wireless-policy)# service-policy input output policy-name
```

These options are available:

- **input**—Assigns the policy-map to all clients in WLAN.
- **output**—Assigns the policy-map to all clients in WLAN.

Step 5 Enable the wireless policy profile.

Example:

```
Device(config-wireless-policy)# no shutdown
```

The specified QoS policy is applied to the WLAN policy profile, and all clients associated with the WLAN receive the defined QoS treatment.

Apply policy profile to policy tag (GUI)

Associate a policy profile with a policy tag to manage network behavior for specific WLANs.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the **Manage Tags** page in the **Policy** tab, click **Add**.

Step 3 In the **Add Policy Tag** window, enter a name and description for the policy tag.

Step 4 Map the required WLAN IDs and WLAN profiles with appropriate policy profiles.

Step 5 Click **Update & Apply to Device**.

The device applies the specified policy profile to the policy tag and controls WLAN behavior based on the configuration.

Apply policy profile to policy tag (CLI)

Associate a policy profile with a policy tag through command-line configuration to enforce desired wireless policy settings.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure policy tag and enter the policy tag configuration mode.

Example:

```
Device(config-policy-tag)# wireless tag policy policy-tag-name
```

Step 3 Map a policy profile to a WLAN profile.

Example:

```
Device(config-policy-tag)# wlan test policy profile-policy-name
```

Step 4 Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example:

```
Device(config-policy-tag)# end
```

Step 5 Display the configured policy tags.

Example:

```
Device# show wireless tag policy summary
```

Note
To view the detailed information of a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

The policy profile is applied to the policy tag. You can verify the configuration through summary and detailed display commands.

Attach policy tag to an AP

Assign a policy tag to a wireless AP to determine its network and functional behavior.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the AP, then enter the AP profile configuration mode.

Example:

```
Device(config)# ap F866.F267.7DFB
```

Step 3 Map a policy tag to the AP.

Example:

```
Device(config-ap-tag)# policy-tag policy-tag-name
```

Step 4 Save the configuration. Exit configuration mode. Return to privileged EXEC mode.

Example:

```
Device(config-ap-tag)# end
```

Step 5 Display the AP details and the tags associated with it.

Example:

```
Device# show ap tag summary
```

The specified AP is associated with the chosen policy tag. You can verify the mapping by displaying the AP details.

Configure custom QoS mapping (CLI)

Create a custom mapping between IP DSCP values and 802.11e user priorities to support Hotspot 2.0 interworking on the WLAN.

The system creates a map between the 802.11e user priorities and the IP differentiated services code point (DSCP) for interworking with IP networks. Enable Hotspot 2.0 on the WLAN to support mapping exception.

Note Custom QoS mapping applies only to Hotspot 2.0.

Specify the mapping by assigning DSCP ranges to individual user priority values, and set exceptions by mapping DSCP values to UP values one-to-one. If you enable a QoS map and do not add custom mappings, the system uses default values.

Note Egress = Downstream = Output; and Ingress = Upstream = Input

The table shows a QoS map, where an AP provides a wireless client with the required mapping from IP DSCP to 802.11e user priority.

Table 33: Default DSCP-Range-to-User Priority Mapping

IP DSCP Range	802.11e User Priority
0-7	0
8-15	1
16-23	2

IP DSCP Range	802.11e User Priority
24-31	3
32-39	4
40-47	5
48-55	6
56-63	7

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile profile-name
```

Step 3 Configure DSCP-to-user priority mapping.

Example:

```
Device(config-ap-profile)# qos-map dscp-to-up-range user-priority up-to-dscp dscp-start
dscp-end
```

You can configure up to eight entries—one for each *user-priority* value. If you do not configure a custom value, the system sends a non-configured value (0xFF) to the AP.

Use the **no** form of this command to disable the configuration. To delete all the custom mappings, use the **no dscp-to-up-range** command.

When you apply custom DSCP to user priority mappings for an AP profile, the system translates the QoS policy correctly for Hotspot 2.0 clients.

Configure a DSCP-to-user priority mapping exception

Define custom exceptions so you can map IP DSCP values to 802.11e user priorities in an AP profile. This provides control over QoS behavior.

When you configure a QoS mapping or exception, the system creates a custom QoS map. The map is sent to the corresponding AP.

If you do not configure DSCP-to-user priority mapping or exception entries, the system uses an empty QoS map.

The table displays exceptions with a one-to-one mapping between DSCP values and user priority values.

Table 34: Default DSCP-Range-to-User Priority Mapping Exceptions

IP DSCP	802.11e User Priority
0	0
2	1
4	1
6	1
10	2
12	2
14	2
18	3
20	3
22	3
26	4
34	5
46	6
48	7
56	7

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile ap profile name
```

Step 3 Configure a DSCP-to-user priority exception.

Example:

```
Device(config-ap-profile)# qos-map dscp-to-up-exception dscp-num user-priority
```

With these settings, your AP profile has customized DSCP-to-user priority mapping exceptions. This configuration provides tailored QoS mapping for wireless clients.

Configure trust upstream DSCP value

Configure the wireless controller to trust the upstream DSCP value, not the user priority. This optimizes end to end QoS marking.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure an AP profile and enter AP profile configuration mode.

Example:

```
Device(config)# ap profile ap profile-name
```

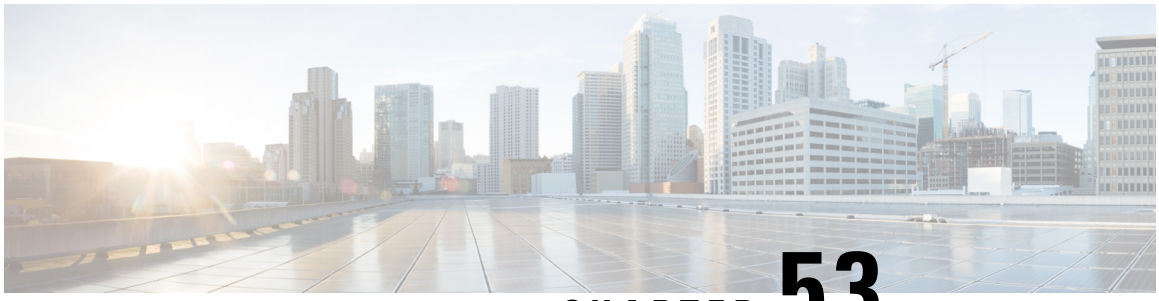
Step 3 Configure the AP to trust upstream DSCP instead of user priority.

Example:

```
Device(config-ap-profile)# qos-map trust-dscp-upstream
```

Use the **no** form of the command to disable the configuration.

The AP profile is set to trust upstream DSCP values, ensuring that client QoS markings are preserved from end to end.



CHAPTER 53

Wireless Auto-QoS

- Auto QoS information, on page 581

Auto QoS information

Wireless Auto QoS automates deployment of wireless QoS features. It has a set of predefined profiles which can be further modified by the customer to prioritize different traffic flows. Auto-QoS matches traffic and assigns each matched packet to qos-groups. This allows the output policy map to put specific qos-groups into specific queues, including into the priority queue.

Autoqos policy configuration

Table 35: Autoqos policy configuration

Mode	Client Ingress	Client Egress	BSSID Ingress	BSSID Egress	Port Ingress	Port Egress	Radio
Voice	N/A	N/A	P3	P4	N/A	P7	ACM on
Guest	N/A	N/A	P5	P6	N/A	P7	
Fastlane	N/A	N/A	N/A	N/A	N/A	P7	edca-parameters fastlane
Enterprise-avc	N/A	N/A	P1	P2	N/A	P7	

P1	AutoQos-4.0-wlan-ET-SSID-Input-AVC-Policy
P2	AutoQos-4.0-wlan-ET-SSID-Output-Policy
P3	platinum-up
P4	platinum
P5	AutoQos-4.0-wlan-Dot1x-Output-Policy
P6	AutoQos-4.0-wlan-GT-SSID-Output-Policy
P7	AutoQos-4.0-wlan-Port-Output-Policy

Configure wireless auto-QoS on profile policy (CLI)

You can enable AutoQoS on a profile policy to automate deployment of wireless QoS features.

You can enable AutoQoS on a profile policy.

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Configure AutoQoS wireless policy.

Example:

```
Device# wireless autoqos policy-profile test-profile mode voice
```

- **enterprise-avc**—Enables AutoQoS Wireless Enterprise AVC Policy.
- **fastlane**—Enable AutoQoS Wireless Fastlane Policy.
- **guest**—Enable AutoQoS Wireless Guest Policy.
- **voice**—Enable AutoQoS Wireless Voice Policy.

Note
AutoQoS MIB attribute does not support full functionality with service policy. Service policy must be configured manually. Currently, there is only support for AutoQoS mode.

What to do next

Note After enabling AutoQoS, we recommend that you wait for a few seconds for the policy to install and then try and modify the AutoQoS policy maps if required; or retry if the modification is rejected.

Disable wireless autoqos

Disable wireless AutoQoS functionality to manually control QoS parameters or troubleshoot QoS-related issues.

Wireless AutoQoS automatically configures QoS settings for optimal wireless performance. You may need to disable it when implementing custom QoS policies or when troubleshooting network issues.

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device> enable
```

Enter your password if prompted.

Step 2 Shut down the policy profile.

Example:

```
Device# shutdown
```

Shuts down the policy profile.

Step 3 Globally disable wireless AutoQoS.

Example:

```
Device# wireless autoqos disable
```

Globally disables wireless AutoQoS.

Step 4 Enable the wireless policy profile.

Example:

```
Device# no shutdown
```

Enables the wireless policy profile.

Note
Disabling Auto QoS does not reset global radio configurations like CAC and EDCA parameters.

Wireless AutoQoS is now disabled globally, allowing for manual QoS configuration or troubleshooting without automatic QoS interference.

Rollback auto-QoS configuration (GUI)

Before you begin

Follow these steps to rollback AutoQoS configuration using the GUI:

Procedure

Step 1 Choose **Configuration** > **Services** > **QoS**.

Step 2 Click **Disable AutoQoS**.

Step 3 Click **Yes** to confirm.

Rollback AutoQoS configuration (CLI)

Before you begin

Note AutoQoS MIB attribute does not support the full functionality with service policy. Currently, there is only support for AutoQoS mode. Service policy must be configured manually.

Follow these steps to rollback AutoQoS configuration:

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Reset AutoQoS configuration.

Example:

```
Device# clear platform software autoqos config template guest
```

- enterprise-avc—Resets AutoQoS Enterprise AVC Policy Template.
- guest—Resets AutoQoS Guest Policy Template.

Clear wireless autoqos policy profile (GUI)

Before you begin

Follow these steps to clear wireless AutoQoS policy profile using the GUI

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click on the **Policy Profile Name**.

Step 3 Go to **QOS and AVC** tab.

Step 4 From the **Auto Qos** drop-down list, choose **None**.

Step 5 Click **Update & Apply to Device**.

Clear wireless autoqos policy profile (CLI)

Clear the AutoQoS configuration from a wireless policy profile to reset or remove QoS settings.

Use this procedure when you need to remove AutoQoS settings from a wireless policy profile, typically during reconfiguration or troubleshooting scenarios.

Procedure

Step 1 Enable privileged EXEC mode.

Example:

```
Device# enable
```

Enter your password if prompted.

Step 2 Shut down the policy profile.

Example:

```
Device# shutdown
```

Step 3 Clear the configured AutoQoS wireless policy.

Example:

```
Device# wireless autoqos policy-profile policy-name mode clear
```

Example:

```
Device# wireless autoqos policy-profile test-profile mode clear
```

Step 4 Enable the wireless policy profile.

Example:

```
Device# no shutdown
```

The AutoQoS configuration is cleared from the specified wireless policy profile, and the profile is re-enabled.

View Auto-QoS on policy profile (CLI)

Before you begin

AutoQoS is supported on the local mode and flex mode. AutoQoS configures a set of policies and radio configurations depending on the template. It is possible to override the service-policy that is configured by AutoQoS. The latest configuration takes effect, with AAA override policy being of highest priority.

Follow these steps to view AutoQoS on policy profile:

Procedure

Step 1 **enable**

Example:

```
Device enable
```

Enables privileged EXEC mode. Enter your password if prompted.

Step 2 **show wireless profile policy detailed** *policy-profile-name*

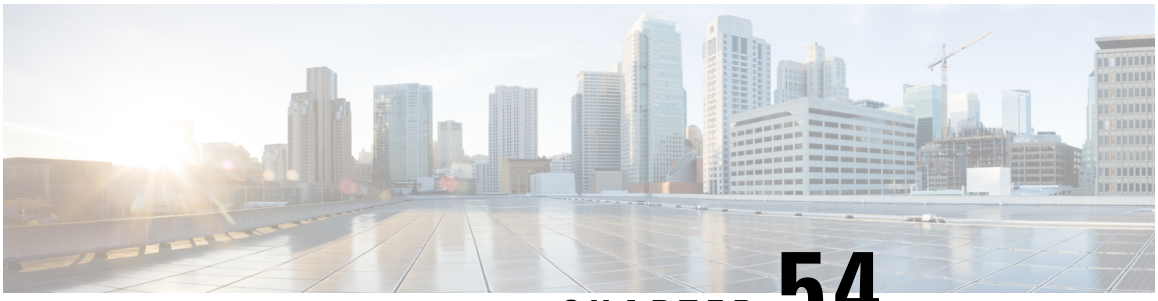
Example:

```
Device# show wireless profile policy detailed testqos
```

Shows policy-profile detailed parameters.

Note
When the AutoQoS mode is configured to voice within the policy profile, any modifications to the global ACM settings require a shutdown and subsequent no shutdown operation on the policy profile to activate the changes. Without performing this shut and no shut cycle, the updated ACM configurations are not applied to the beacon frames. This causes the beacon to continue broadcasting the old ACM values.

This process ensures that the revised ACM settings are correctly propagated and reflected throughout the wireless environment, thereby maintaining accurate call admission control enforcement.



CHAPTER 54

Native Profiling

- Native profiling, on page 587

Native profiling

Native profiling is a network security feature that

- profiles devices based on HTTP and DHCP to identify end devices on the network
- configures device-based policies and enforces these policies per user or per device policy on the network, and
- allows profiling of mobile devices and basic onboarding of the profiled devices to a specific VLAN.

Native profiling characteristics

Policies assign ACL and QoS or configure session timeouts.

The policies are defined based on the following attributes:

- User group or user role
- Device type such as Windows clients, smartphones, tablets, and so on
- Service Set Identifier (SSID)
- Location, based on the access point group that the end point is connected to
- Time of the day
- Extensible Authentication Protocol (EAP) type, to check what EAP method that the client is getting connected to

When a wireless client joins an access point, certain QoS policies get enforced on the access point. One such feature is the native profiling for both upstream and downstream traffic at AP. The native profiling feature when clubbed with AAA override supports specific set of policies based on the time of day and day of week. The AAA override then applies these policies coming from a RADIUS server to the access point.

Let's consider a use case of time of the day in conjunction with user role. Usually, the user role is used as an extra matching criteria along with the time of day. You can club the time of day usage with any matching criteria to get the desired result. The matching will be performed when the client joins the embedded wireless controller.

You can configure policies as two separate components:

- Defining policy attributes as service templates that are specific to clients joining the network and applying policy match criteria
- Applying match criteria to the policy.

Note Before proceeding with the native profile configuration, ensure that HTTP Profiling and DHCP Profiling are enabled.

To configure Native Profiling, use one of the following procedures:

- Create a service template
- Create a class map

Note You can apply a service template using either a class map or parameter map.

- Create a parameter-map and associate the service template to parameter-map
  - Create a policy map
    1. If class-map has to be used: Associate the class-map to the policy-map and associate the service-template to the class-map.
    2. If parameter-map has to be used: Associate the parameter-map to the policy-map
  - Associate the policy-map to the policy profile.

Create a class map (GUI)

Procedure

Step 1 Click **Configuration** > **Services** > **QoS**.

Step 2 In the **QoS – Policy** area, click **Add** to create a new QoS Policy or click the one you want to edit.

Step 3 Add **Add Class Map** and enter the details.

Step 4 Click **Save**.

Step 5 Click **Update and Apply to Device**.

Create a class map (CLI)

Note Configuration of class maps via CLI offer more options and can be more granular than GUI.

Before you begin

Follow these steps to create a class map using CLI commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-any cls_user
```

Step 3 Specify the class map attribute filter criteria.

Example:

```
Device(config-filter-control-classmap)# match username ciscoise
```

Step 4 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-any cls_userrole
```

Step 5 Specify the class map attribute filter criteria.

Example:

```
Device(config-filter-control-classmap)# match user-role engineer
```

Step 6 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-any cls_oui
```

Step 7 Specify the class map attribute filter criteria.

Example:

```
Device(config-filter-control-classmap)# match oui 48.f8.b3
```

Step 8 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-any cls_mac
```

Step 9 Specify the class map attribute filter criteria.

Example:

```
Device(config-filter-control-classmap)# match mac-address 0040.96b9.4a0d
```

Step 10 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-any cls_devtype
```

Step 11 Specify the class map attribute filter criteria.

Example:

```
Device(config-filter-control-classmap)# match device-type windows
```

Step 12 Specify the class map type and name.

Example:

```
Device(config)# class-map type control subscriber match-all match_tod
```

Step 13 Specify a match to the time of day.

Example:

```
Device(config-filter-control-classmap)# match join-time-of-day 10:30 12:30
```

Here, join time is considered for matching. For example, if the match filter is set from 11:00 am to 2:00 pm, a device joining at 10:59 am is not considered, even if it acquires credentials after 11:00 am.

Here,

start-time and *end-time* specifies the 24-hour format.

Use the **show class-map type control subscriber name** *name* command to verify the configuration.

Note
You should also disable AAA override for this command to work.

Step 14 Match name using the device type.

Example:

```
Device(config-filter-control-classmap)# match device-type android
```

Type a question mark (?) after the device type and select the device from the list.

Note
You should enable the device classifier for the device list to be populated.

Create a service template (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Local Policy**.

Step 2 On the **Local Policy** page, **Service Template** tab, click **ADD**.

Step 3 In the **Create Service Template** window, enter the following parameters:

- **Service Template Name:** Enter a name for the template.
- **VLAN ID:** Enter the VLAN ID for the template. Valid range is between 1 and 4094.
- **Session Timeout (secs):** Sets the timeout duration for the template. Valid range is between 1 and 65535.
- **Access Control List:** Choose the Access Control List from the drop-down list.
- **Ingress QOS:** Choose the input QoS policy for the client from the drop-down list
- **Egress QOS:** Choose the output QoS policy for the client from the drop-down list.

Step 4 Click **Save & Apply to Device**.

Create a service template (CLI)

Before you begin

Follow these steps to create a service template using CLI commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter service template configuration mode.

Example:

```
Device(config)# service-template svc1
```

Step 3 Specify the access list to be applied.

Example:

```
Device(config-service-template)# access-group acl-auto
```

Step 4 Specify VLAN ID.

Example:

```
Device(config-service-template)# vlan 10
```

Valid range is from 1-4094.

Step 5 Specify session timeout value for a service template.

Example:

```
Device(config-service-template)# absolute-timer 1000
```

Valid range is from 1-65535.

Step 6 Configure an input QoS policy for the client.

Example:

```
Device(config-service-template)# service-policy qos input in_qos
```

Step 7 Configure an output QoS policy for the client.

Example:

```
Device(config-service-template)# service-policy qos output out_qos
```

Create a parameter map (CLI)

Before you begin

Follow these steps to create a parameter map:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the parameter map type and name.

Example:

```
Device(config)# parameter-map type subscriber attribute-to-service param
```

Step 3 Specify the parameter map attribute filter criteria.

Example:

```
Device(config-parameter-map-filter)# 1 map device-type eq "windows"
mac-address eq 3c77.e602.2f91 username eq "cisco"
```

Multiple filters are used in the example provided here.

Step 4 Specify the service template and its precedence.

Example:

```
Device(config-parameter-map-filter-submode)# 1 service-template svc1 precedence 150
```

Create a policy map (GUI)

Before you begin

Follow these steps to create a policy map using the GUI:

Procedure

Step 1 Choose **Configuration** > **Security** > **Local Policy** > **Policy Map** tab.

Step 2 Enter a name for the Policy Map in the **Policy Map Name** text field.

Step 3 Click **Add**

Step 4 Choose the service template from the **Service Template** drop-down list.

Step 5 For the following parameters select the type of filter from the drop-down list and enter the required match criteria

- Device Type
- User Role
- User Name
- OUI
- MAC Address

Step 6 Click **Add Criteria**

Step 7 Click **Update & Apply to Device**.

Create a policy map (CLI)

Create and configure a policy map to control subscriber behavior and apply service templates in wireless environments.

Policy maps define how the system responds to specific events and conditions. This procedure creates a subscriber control policy map that can be applied to wireless profiles for client management and service provisioning.

Before you begin

Before removing a policy map or parameter map, you should remove it from the target or shut down the WLAN profile or delete the session.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the policy map type.

Example:

```
Device(config)# policy-map type control subscriber policy-map-name
```

Example:

```
Device(config)# policy-map type control subscriber polmap5
```

Step 3 Specify the match criteria to the policy map.

Example:

```
Device(config-event-control-policymap)# event identity-update match-all
```

Step 4 You can apply a service template using either a class map or a parameter map, as shown here.

- Configure the local profiling policy class map number and specify how to perform the action.
- Activate the service template.
- Map an identity-update attribute to an auto-configured template.

Example:

The following example shows how a class-map with a service-template has to be applied:

```
Device(config-class-control-policymap)# class-num class class-map-name do-until-failure
Device(config-action-control-policymap)# action-index activate service-template
service-template-name
```

Example:

The following example shows how a parameter map has to be applied (service template is already associated with the parameter map 'param' while creating it):

```
Device(config-action-control-policymap)# action-index map attribute-to-service table
parameter-map-name
```

Step 5 Exit configuration mode.

Example:

```
Device(config-action-control-policymap)# end
```

Step 6 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 7 Configure a wireless policy profile.

Example:

```
Device(config)# wireless profile policy wlan-policy-profile-name
```

Caution
Do not configure aaa-override for native profiling under a named wireless profile policy. Native profiling is applied at a lower priority than AAA policy. If aaa-override is enabled, the AAA policies will override native profile policy.

Step 8 Add a description for the policy profile.

Example:

```
Device(config-wireless-policy)# description profile-policy-description
```

Example:

```
Device(config-wireless-policy)# description "default policy profile"
```

Step 9 Configure DHCP TLV caching on a WLAN.

Example:

```
Device(config-wireless-policy)# dhcp-tlv-caching
```

Step 10 Configure client HTTP TLV caching on a WLAN.

Example:

```
Device(config-wireless-policy)# http-tlv-caching
```

Step 11 Configure the subscriber policy name.

Example:

```
Device(config-wireless-policy)# subscriber-policy-name policy-name
```

Example:

```
Device(config-wireless-policy)# subscriber-policy-name polmap5
```

Step 12 Configure a VLAN name or VLAN ID.

Example:

```
Device(config-wireless-policy)# vlan vlan-id
```

Example:

```
Device(config-wireless-policy)# vlan 1
```

Step 13 Save the configuration.

Example:

```
Device(config-wireless-policy)# no shutdown
```

The policy map is created and configured with the specified parameters, and the wireless policy profile is activated with the associated subscriber policy.

Configure native profiling in local mode (CLI)

To enable native profiling for wireless users in local mode by updating the policy profile configuration.

To configure native profiling in the local mode, you must follow the steps described in Create a policy map (CLI), on page 593 . In the policy profile, you must enable central switching as described in the step given below in order to configure native profiling.

Procedure

Enable central switching.

Example:

```
Device(config-wireless-policy)# central switching
```

Native profiling is now enabled in local mode. The device uses central switching, allowing profiling information to be collected and used for network management.

Verify native profile configuration

Use these **show** commands to verify the native profile configuartion:

```
Device# show wireless client device summary

Active classified device summary
```

```
MAC Address       Device-type                      User-role
Protocol-map
-------------------------------------------------------------------------------------------------
1491.82b8.f94b    Microsoft-Workstation            sales
            9
1491.82bc.2fd5    Windows7-Workstation             sales
           41


Device# show wireless client device cache

Cached classified device info

MAC Address       Device-type                      User-role
Protocol-map
-------------------------------------------------------------------------------------------------
2477.031b.aa18    Microsoft-Workstation
            9
30a8.db3b.a753    Un-Classified Device
            9
4400.1011.e8b5    Un-Classified Device
            9
980c.a569.7dd0    Un-Classified Device
```

```
Device# show wireless client mac-address 4c34.8845.e32c detail | s
Session Manager:
  Interface :
  IIF ID            : 0x90000002
  Device Type       : Microsoft-Workstation
  Protocol Map      : 0x000009
  Authorized        : TRUE
  Session timeout   : 1800
  Common Session ID: 783802090000000174BF2B5B9
  Acct Session ID   : 0
  Auth Method Status List
   Method : MAB
    SM State         : TERMINATE
    Authen Status    : Success
  Local Polices:
   Service Template : wlan_svc_C414.3CCA.0A51 (priority 254)
    Absolute-Timer   : 1800
  Server Polices:
  Resultant Policies:
  Filter-ID         : acl-auto
  Input QOS         : in_qos
  Output QOS        : out_qos
  Idle timeout      : 60 sec
  VLAN               : 10
  Absolute-Timer    : 1000
```

Use this **show** command to verify the class map details for a class map name:

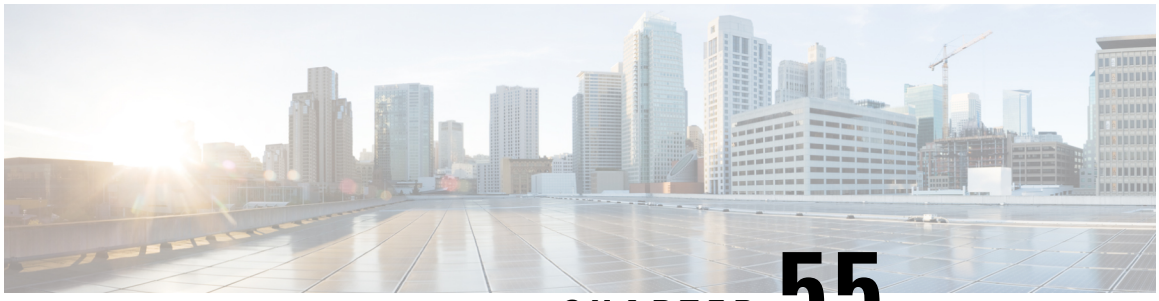
```
Device# show class-map type control subscriber name test
Class-map               Action                        Exec  Hit  Miss  Comp
---------                ------                        ----  ---  ----   ---
match-any test     match day Monday                     0     0    0     0
match-any test     match join-time-of-day 8:00 18:00    0     0    0     0
Key:
  "Exec" - The number of times this line was executed
  "Hit"  - The number of times this line evaluated to TRUE
  "Miss" - The number of times this line evaluated to FALSE
  "Comp" - The number of times this line completed the execution of its
       condition without a need to continue on to the end
```



PART VIII

CleanAir

- Cisco CleanAir, on page 599
- Spectrum Intelligence, on page 617



CHAPTER 55

Cisco CleanAir

- Cisco ISE systems, on page 599
- Configure CleanAir, on page 605
- Verify CleanAir parameters, on page 613
- CleanAir FAQs, on page 615

Cisco ISE systems

A Cisco ISE system is a wireless spectrum management solution that

- proactively monitors and manages challenges caused by interference in a shared wireless spectrum
- uses Cisco CleanAir-enabled access points and controllers to detect and analyze native and foreign interference sources, and
- enables both manual and automated remediation to optimize wireless network operation.

Table 36: Feature history for CleanAir

Release	Feature	Feature Information
Cisco IOS XE 17.13.1	EDRRM Support for 6-GHz Band Radio	The Event-Driven Radio Resource Management (EDRRM) is enabled in the 6-GHz band radio of AP.

Components of Cisco ISE systems

Cisco CleanAir systems are comprised of several key hardware and software elements:

- Cisco CleanAir-**enabled APs**: These devices actively scan the wireless spectrum for all active transmitters.
- **Wireless controllers**: , Cisco Embedded Wireless Controller on Catalyst Access Points, and .
- **Management platforms**: provide centralized management and visualization of spectrum and interference data forwarded by the controllers.

How Cisco CleanAir systems detect and respond to interference

1. **Always-On spectrum awareness**: Cisco ISE-enabled APs scan the industrial, scientific, and medical (ISM) bands, identifying and evaluating all radio devices that may cause interference.
2. **Unified control and interference intelligence**: The controller embedded wireless controller controls the access points.
3. **Actionable Insight for Every Interferer**: For every device operating in the unlicensed band, Cisco ISE provides information about what it is, how it is impacting your wireless network, and what actions you or your network should take. It simplifies RF.

Benefits of Cisco ISE systems

- **Simplifies spectrum management**: Provides visibility into interference without requiring advanced RF expertise.
- **Improves network reliability**: Automatically detects and resolves issues affecting Wi-Fi performance.
- **Supports advanced WLAN services**: Ensures quality for voice over wireless and IEEE 802.11 operations by minimizing RF interference.

Wireless LAN systems use unlicensed 2.4-GHz and 5-GHz ISM bands, which are also shared by devices such as microwave ovens, cordless phones, and Bluetooth equipment. These non-Wi-Fi devices can disrupt network services including voice-over-wireless and IEEE 802.11 communications. Cisco CleanAir systems detect such interference and help mitigate its effects.

Cisco ISE-related terms

Table 37: Cisco CleanAir-related terms

Term	Description
AQI	Air Quality Index. The AQI indicates air quality based on RF interference. An AQI of zero is considered poor, and an AQI greater than eighty-five is considered good.
AQR	Air Quality Report. An AQR shows total interference from all sources, including major categories. The system sends an AQR every 15 minutes and every 30 seconds in Rapid mode.
DC	Duty Cycle. Indicates the percentage of time your device uses the channel.
EDRRM	Event-Driven RRM. When needed, EDRRM enables your access point to skip normal RRM intervals and immediately change channels.
IDR	Interference Device Reports that an AP sends to the embedded wireless controller.
ISI	Interference Severity Index. The ISI is an indicator of the severity of the interference.
RSSI	Received Signal Strength Indicator. RSSI measures the power in a received signal. It indicates the level detected by an access point from an interfering device.

Cisco CleanAir components

Summary

Cisco CleanAir technology uses a coordinated set of hardware and software components to detect, report, and address wireless interference. Each part of the system contributes to maintaining optimal air quality and reliable wireless performance.

The key components involved in the process are:

- Cisco CleanAir-enabled AP: Collects information about Wi-Fi interference sources processes it. The AP sends the Air Quality Report (AQR) and Interference Device Report (IDR) to the embedded wireless controller.
- Controller: Controls and configures CleanAir-capable access points, collects and processes spectrum data, provides user interfaces (GUI and CLI), and detects, merges, and mitigates interference devices. The also detects, merges, and mitigates interference devices using RRM TPC and DCA For details, see Interference Device Merging.

CleanAir technology ensures wireless networks remain reliable by automatically detecting sources of interference and providing tools for administrators to analyze and respond to RF threats.

Workflow

These stages describe the operation of Cisco CleanAir components in the system.

1. Configuration: Configures Cisco CleanAir capabilities on the APs.
2. Detection: Cisco CleanAir-enabled APs continuously scan the wireless environment for interference sources and assess air quality.
3. Reporting: APs generate reports such as Air Quality Reports (AQRs) and Interference Device Reports (IDRs), which summarize detected interference and air quality indexes.
4. Aggregation: Wireless controllers collect AQRs and IDRs from all APs, process the data, and store it in relevant databases.
5. Analysis and Response: Controllers use built-in mechanisms (like Radio Resource Management) to mitigate interference and maintain optimal RF conditions.
6. Advanced Services: MSE tracks interference devices over time and location, merges data from multiple controllers, and provides comprehensive threat detection with adaptive Wireless Intrusion Prevention System (WIPS).
7. Specialized Troubleshooting (optional): For in-depth RF analysis, an administrator can connect a CleanAir-enabled AP directly to a PC running Cisco Spectrum Expert.

Result

The Cisco CleanAir system delivers end-to-end automated detection, analysis, and mitigation of wireless interference, enabling proactive network management and high-quality wireless performance.

Prerequisites for CleanAir

You can configure Cisco CleanAir only on CleanAir-enabled access points.

Only Cisco CleanAir-enabled access points in these access point modes can perform Cisco CleanAir spectrum monitoring:

- **Local** —In this mode, each Cisco CleanAir-enabled access point radio provides air quality and interference detection reports for the current operating channel only. An AP measures air quality and interference only when it is not transmitting Wi-Fi frames. When channel utilization is high, CleanAir detects less interference.
- **FlexConnect** —When a FlexConnect access point is connected to the embedded wireless controller , its Cisco CleanAir functionality is identical to local mode.
- **Monitor** —When Cisco CleanAir is enabled in monitor mode, the access point provides air quality and interference detection reports for all monitored channels.
  - **All** —All channels
  - **DCA** —Channel selection governed by the DCA list
  - **Country** —All channels are legal within a regulatory domain
- **SE-Connect** : Use this mode to connect Spectrum Expert on a Microsoft Windows XP or Vista PC to a Cisco CleanAir-enabled access point and display detailed spectrum data. The application connects directly to the AP and bypasses the device. In SE-Connect mode, the AP does not provide Wi-Fi, RF, or spectrum data; it does not serve clients. Use this mode only for remote troubleshooting. You can have up to three active Spectrum Expert connections.

Only Cisco Catalyst 3850 and Switches can function as Mobility Agents.

Cisco Catalyst 3850 Switches, and Cisco 5760 Wireless LAN Controllers can function as Mobility Controllers.

Restrictions for CleanAir

Access points in monitor mode do not transmit Wi-Fi traffic or 802.11 packets. They are excluded from radio resource management (RRM) planning and are not included in the neighbor AP list. IDR clustering depends on the device ’s ability to detect neighboring in-network APs. Correlating interference device detections from multiple APs is limited between monitor-mode APs.

-
- In the 4800 AP, slot 1 is dedicated to 5 GHz and cannot be moved to monitor mode. Slot 0 is an XOR slot and can be set to monitor mode, 2.4 GHz, or 5 GHz. Slot 2 is dedicated to monitor mode and will operate in 5 GHz. When AP monitor mode is enabled, slot 2 is disabled because a monitor radio is already available for both 2.4 and 5 GHz. The 3700 AP has dedicated 2.4 GHz (slot 0) and 5 GHz (slot 1) radios.

Do not connect APs in SE connect mode directly to any physical port on the controller.

CleanAir is not supported where the channel width is 160 MHz.

Difference between Wi-Fi chip-based RF Management and Cisco CleanAir

Feature	Wi-Fi Chip–Based RF Management	Cisco CleanAir
Noise identification	Reports broad, often vague RF noise without pinpointing sources	Precisely identifies interference type, source, potential impact to a WLAN

Feature	Wi-Fi Chip–Based RF Management	Cisco CleanAir
Noise measurement method	Relies on long-term averaging, which smooths out important details and reduces resolution of measurements	Captures real-time events with full granularity
Response style	Primarily reacts after interference is noticed	Enables proactive, context-aware mitigation allowing for intelligent, proactive decisions.
Transient interference handling	Averaging measurements reduces resolution of measurements, and disruptive signals do not appear to need mitigation.	Flags brief, interference events immediately

Cisco CleanAir .

Note Spectrum event-driven RRM can be triggered only by Cisco CleanAir-enabled access points in local mode.

Persistence interference

A persistent device avoidance is a wireless interference mitigation feature that

- identifies and continuously tracks intermittent interference sources such as microwave ovens,
- maintains channel avoidance for affected APs even if the interference is not constantly present, and
- automatically updates Radio Resource Management (RRM) whenever the interfering device or related access points are moved.

For example, if a microwave oven operates in a break room only for a few minutes at a time, Cisco CleanAir identifies the microwave as an interference source, locates the affected portion of the band and APs most severely affected, and directs RRM to avoid those channels persistently. Even if the microwave is not active all day, the system maintains avoidance as long as the device is periodically detected. If you move either the microwave oven or the surrounding access points, the algorithm updates RRM automatically

Classes of persistent interference sources

Certain devices, once detected, are likely to remain in place and may cause ongoing interference. Cisco CleanAir can bias the affected channel in Radio Resource Management (RRM) so that CleanAir "remembers" the potential for interference. See Radio Resource Management White Paper.

These device classes qualify as persistent interference sources:

- Microwave ovens
- Outdoor Ethernet bridges
- WiMax fixed and mobile devices
- Motorola Canopy systems

Special case: Bluetooth devices

Bluetooth devices are detected and reported as interference only when actively transmitting. They often enter power-save modes, so interference may only be noticed during active streaming of data or voice.

Persistent devices detection

CleanAir-capable Monitor Mode AP collects information about persistent devices on all configured channels and store the information in embedded wireless controller. Local or Bridge mode AP detects interference devices on the serving channels only.

Persistent device avoidance

When a Persistent Device (PD) is detected in the CleanAir module, it is reported to the RRM module on the MA. This information is used in the channel selection by the subsequent EDRRM Event Driven RRM (ED-RRM) signal sent to the RRM module.

Spontaneous interference

Spontaneous interference is interference that appears suddenly on a network and may jam one or more channels. With Cisco CleanAir spectrum, event-driven RRM, you can set a threshold for air quality, abbreviated as AQ. If air quality drops below this threshold, the system immediately changes the affected AP's channel. Although most RF management systems avoid interference, they require time for information to spread before channels update. Cisco CleanAir uses AQ measurements to continually check the spectrum and can switch channels within **30 seconds**.

For example, if your AP detects interference from a device such as a video camera, it switches channels within **30 seconds** after the interference begins. Cisco CleanAir also helps you identify and locate the interference source so you can take permanent steps to resolve the issue later.

EDRRM and AQR update mode

An EDRRM and AQR update mode is a wireless network feature set that

- enables APs experiencing severe interference ("in distress") to bypass normal RRM intervals and immediately change channels,
- requires CleanAir technology for monitoring and reporting classified interference devices, and
- provides fast action and remote configuration through control messages and local spectrum management.
- **EDRRM (Event Driven Radio Resource Management):** A function that allows an AP to react instantly to excessive interference by changing channels immediately instead of waiting for scheduled RRM intervals.
- **AQR (Air Quality Reporting) update mode**: A process by which CleanAir APs monitor air quality and report classified interference devices every 15 minutes; when interference causes substantial air quality degradation, EDRRM is triggered.

If a classified interference device severely degrades the air quality on an active channel, EDRRM triggers an immediate channel change so that no clients are stranded on an unusable channel.

Think of a city's scheduled traffic lights as the normal RRM intervals, allowing cars (data) to move in an orderly, timed way. Normally, everyone waits for the green light, moving only at the set intervals. EDRRM is like having an emergency override for ambulances (APs in distress): if there's a medical emergency (severe

interference), the control center can change the lights to green immediately—regardless of the normal schedule—so the ambulance can take an alternate, clear route (new channel). Air Quality Reporting (AQR) acts like real-time traffic cameras detecting and reporting sudden blockages (classified interference devices) so that the emergency override can be triggered if road conditions (channel quality) deteriorate unexpectedly.

Restrictions for EDRRM and AQR update mode

- EDRRM is not enabled by default; you must first enable CleanAir and then enable EDRRM.
- If an interfering device is operating on an active channel and causes enough AQ degradation to trigger an EDRRM, then no clients will be able to use that channel or the AP. You must remove the AP from the channel.

Configure CleanAir

Enable CleanAir for the 2.4-GHz band (GUI)

Enable CleanAir on 2.4-GHz band radios for real-time interference detection.

Use this procedure to activate Cisco CleanAir technology on APs operating in the 2.4-GHz spectrum using the GUI.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 On the **CleanAir** window, click the **2.4 GHz Band** > **General** tab.

Step 3 Check the **Enable CleanAir** checkbox.

Step 4 Click **Apply**.

CleanAir is enabled on all selected 2.4-GHz radios, and spectrum analysis begins for those devices.

Enable CleanAir for the 2.4-GHz band (CLI)

Enable CleanAir technology on APs operating in the 2.4-GHz band radios for real-time interference detection using commands.

Procedure

Enable the CleanAir feature on the 802.11b network.

Example:

```
Device(config)#ap dot11 24ghz cleanair
Device(config)#no ap dot11 24ghz cleanair
```

Run the **no** form of this command to disable CleanAir on the 802.11b network.

CleanAir is enabled on all selected 2.4-GHz radios, and spectrum analysis begins for those devices.

Configure a CleanAir alarm for 2.4-GHz air-quality and devices (CLI)

Set up alarms that notify you when air quality drops or when certain device types are detected on the 2.4-GHz band using commands.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the alarm for the threshold value for air-quality for all the 2.4-GHz devices.

Example:

```
Device(config)# ap dot11 24ghz cleanair alarm air-quality threshold threshold_value
```

Add the **no** form of this command to disable the alarm.

Step 3 Configure the alarm for the 2.4-GHz devices using this command.

Example:

```
Device(config)# ap dot11 24ghz cleanair alarm device {bt-discovery | bt-link | canopy |
cont-tx | dect-like | fh | inv | jammer | mw-oven | nonstd | report | superag | tdd-tx |
video | wimax-fixed | wimax-mobile | xbox | zigbee}
```

Add the **no** form command to disable the alarm.

- bt-discovery: Bluetooth Discovery and bt-link: Bluetooth Link.
- canopy: Canopy devices and cont-tx: Continuous Transmitter.
- dect-like: Digital Enhanced Cordless Communication (DECT)-like phone and fh: 802.11 frequency hopping devices.
- inv: Devices using spectrally inverted WiFi signals and jammer: Jammer.
- mw-oven: Microwave oven and nonstd: Devices using non standard Wi-Fi channels.
- report: Interference device reporting and superag: 802.11 SuperAG devices.
- tdd-tx: TDD Transmitter and video: Video cameras.
- wimax-fixed: WiMax Fixed and wimax-mobile: WiMax Mobile.
- xbox: Xbox and zigbee: 802.15.4 devices.

Step 4 Return to the privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure interference reporting for a 2.4-GHz device (GUI)

Configure which types of interference are detected by a 2.4-GHz device using the CleanAir GUI. This task guides you through selecting and applying specific interference types to monitor.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 Click the **2.4 GHz Band** tab.

Step 3 Choose the interference types and add them to the **Interference Types to detect** section.

These interference types are available:

- BLE Beacon: Bluetooth low energy beacon and Bluetooth Discovery.
- Bluetooth Link and Canopy.
- Continuous Transmitter and DECT-like Phone: Digital Enhanced Cordless Technology phone.
- 802.11 FH: 802.11 frequency hopping device and WiFi Inverted: Device using spectrally inverted Wi-Fi signals.
- Jammer and Microwave Oven.
- WiFi Invalid Channel: Device using nonstandard Wi-Fi channels and TDD Transmitter.
- Video Camera and SuperAG: 802.11 SuperAG device.
- WiMax Mobile and WiMax Fixed.
- 802.15.4, Microsoft Device and SI_FHSS.

Step 4 Click **Apply**.

Configure interference reporting for a 2.4-GHz device (CLI)

Configure a 2.4-GHz device to report interference using commands. This process enables devices to detect and provide information about various sources of wireless interference.

Procedure

Step 1 Configure the 2.4-GHz interference devices to report to the device using the command.

Example:

```
Device(config)# ap dot11 24ghz cleanair device {ble-beacon | bt-discovery | bt-link | canopy
 | cont-tx | dect-like | fh | inv | jammer | mw-oven | nonstd | report | superag | tdd-tx
| video | wimax-fixed | wimax-mobile | xbox | zigbee | alarm}
```

Run the **no** form of this command to disable the configuration.

These are the list of the keyword descriptions:

- bt-discovery: Bluetooth discovery and bt-link: Bluetooth link.
- canopy: Canopy device and cont-tx: Continuous transmitter.
- dect-like: Digital Enhanced Cordless Communication-like phone and fh: 802.11-frequency hopping device.
- inv: Device using spectrally inverted Wi-Fi signals and jammer: Jammer.
- mw-oven: Microwave oven, nonstd: Device using nonstandard Wi-Fi channels and report: Interference device reporting.
- superag: 802.11 SuperAG device, tdd-tx: TDD transmitter and video: Video camera.
- wimax-fixed: WiMax Fixed and wimax-mobile: WiMax Mobile.
- microsoft xbox: Microsoft Xbox device and zigbee: 802.15.4 device.

Step 2 Return to the privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Enable CleanAir for the 5-GHz band (GUI)

Enable CleanAir to improve 5-GHz wireless interference detection and spectrum analysis using the GUI.

Perform this task when you need enhanced monitoring and automated mitigation of interference on 5-GHz wireless networks.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 On the **CleanAir** page, choose the **5 GHz Band** > **General** tab.

Step 3 Check the **Enable CleanAir** checkbox.

Step 4 Click **Apply**.

CleanAir is enabled for the 5-GHz band, enhancing the detection and mitigation of wireless interference.

Enable CleanAir for the 5-GHz band (CLI)

Enable the CleanAir feature to monitor and mitigate wireless interference on 5-GHz wireless networks using commands.

Procedure

Enable the CleanAir feature on a 802.11a network.

Example:

```
Device(config)# ap dot11 5ghz cleanair
```

Run the **no** form of this command to disable CleanAir on the 802.11a network.

CleanAir is enabled for the 5-GHz band, enhancing the detection and mitigation of wireless interference.

Configure interference reporting for a 5-GHz device (GUI)

Enable detection and reporting of selected interference types for better network performance.

Use this procedure when you need to monitor wireless interference on 5-GHz band devices using CleanAir technology.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 Click the **5 GHz Band** tab.

Step 3 Choose the interference types and add them to the **Interference Types to detect** section.

These interference types are available:

- Canopy
- Continuous Transmitter
- DECT-like Phone—Digital Enhanced Cordless Technology phone
- 802.11 FH—802.11 frequency hopping device
- WiFi Inverted—Device using spectrally inverted Wi-Fi signals
- Jammer
- WiFi Invalid Channel—Device using nonstandard Wi-Fi channels
- SuperAG—802.11 SuperAG device
- TDD Transmitter
- WiMax Mobile
- WiMax Fixed

- Video Camera

Step 4 Click **Apply**.

Your 5-GHz device now detects and reports the selected interference types.

Configure interference reporting for a 5-GHz device (CLI)

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a 5-GHz interference device to report to the **ap dot11 5ghz cleanair device**{**canopy | cont-tx | dect-like | inv | jammer | nonstd | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile**} command.

Example:

```
Device(config)#ap dot11 5ghz cleanair device canopy
Device(config)#ap dot11 5ghz cleanair device cont-tx
Device(config)#ap dot11 5ghz cleanair device dect-like
Device(config)#ap dot11 5ghz cleanair device inv
Device(config)#ap dot11 5ghz cleanair device jammer
Device(config)#ap dot11 5ghz cleanair device nonstd

Device(config)#ap dot11 5ghz cleanair device report
Device(config)#ap dot11 5ghz cleanair device superag
Device(config)#ap dot11 5ghz cleanair device tdd-tx
Device(config)#ap dot11 5ghz cleanair device video
Device(config)#ap dot11 5ghz cleanair device wimax-fixed
Device(config)#ap dot11 5ghz cleanair device wimax-mobile
Device(config)#ap dot11 5ghz cleanair device si_fhss
Device(config)#ap dot11 5ghz cleanair device alarm
```

Device. Run the **no** form of this command to disable interference device reporting.

This is a list of the keyword descriptions:

- **canopy**—Canopy device

- **cont-tx**—Continuous transmitter
- **dect-like**—Digital Enhanced Cordless Communication-like phone
- **fh**—802.11-frequency hopping device
- **inv**—Device using spectrally-inverted Wi-Fi signals
- **jammer**—Jammer
- **nonstd**—Device using nonstandard Wi-Fi channels
- **superag**—802.11 SuperAG device
- **tdd-tx**—TDD transmitter
- **video**—Video camera
- **wimax-fixed**—WiMax fixed
- **wimax-mobile**—WiMax mobile

Step 3 Return to the privileged EXEC mode.

Example:

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure event driven RRM for a CleanAir event (GUI)

Configure event driven Radio Resource Management (RRM) to respond to CleanAir-detected interference.

Use this feature to optimize channel assignment automatically when a CleanAir-enabled access point detects significant interference.

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM**.
The **Radio Resource Management** window is displayed.

Step 2 Click the **DCA** tab.

Step 3 In the **Event Driven RRM** section, check the **EDRRM** check box to run RRM when CleanAir-enabled AP detects a significant level of interference.

Step 4 Configure the **Sensitivity Threshold** level at which RRM has to be invoked from the following options:

- **Low**: Represents a decreased sensitivity to changes in the environment and its value is set at 35.
- **Medium**: Represents medium sensitivity to changes in the environment at its value is set at 50.
- **High**: Represents increased sensitivity to changes in the environment at its value is set at 60.
- **Custom**: If you choose this option, you must specify a custom value in the **Custom Threshold** box.

Step 5 To configure rogue duty cycle, check the **Rogue Contribution** check box and then specify the **Rogue Duty-Cycle** in terms of percentage. The default value of rogue duty cycle is 80 percent.

Note
Rogue Contribution is a new component in ED-RRM functionality. It enables ED-RRM to respond to Rogue Channel Utilization, which operates independently from CleanAir metrics. Rogue Duty Cycle is derived from standard off-channel RRM metrics and triggers a channel change when neighboring rogue interference is detected. Since this process relies on RRM metrics rather than CleanAir, the timing depends on normal 180-second off-channel intervals, with a maximum delay of three minutes. Rogue Contribution is configured separately from CleanAir ED-RRM and is disabled by default. This feature allows each AP to respond to Wi-Fi interference originating outside the local network, with measurements taken at every individual AP.

Rogue Contribution is a new component in ED-RRM functionality. It enables ED-RRM to respond to detected Rogue Channel Utilization, which functions separately from CleanAir metrics.

Step 6 Save the configuration.

The controller now automatically triggers RRM based on CleanAir or optionally, rogue channel utilization events.

Configure EDRRM for a CleanAir event (CLI)

Enable automated, event-driven dynamic RRM adjustments in response to CleanAir events.

EDRRM allows the AP to adjust its channel management dynamically in response to detected interference or CleanAir events. This ensures better wireless performance and mitigation of interference.

Before you begin

Verify the AP supports CleanAir and EDRRM features.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# conf t
```

Run the **no** form of this command to disable EDRRM.

Step 2 Enable EDRRM CleanAir event.

Example:

```
Device (config)#ap dot11 24ghz rrm channel cleanair-event
```

```
Device(config)#no ap dot11 24ghz rrm channel cleanair-event
```

Run the **no** form of this command to disable EDRRM.

Step 3 Configure the EDRRM sensitivity of the CleanAir event.

Example:

```
Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high
```

This a list of the keyword descriptions:

- High—Specifies the most sensitivity to non-Wi–Fi interference as indicated by the AQ value.
- Low—Specifies the least sensitivity to non-Wi–Fi interference as indicated by the AQ value.
- Medium—Specifies medium sensitivity to non-Wi–Fi interference as indicated by the AQ value.

EDRRM CleanAir event detection and response is enabled with your selected sensitivity, improving dynamic channel management during interference events.

Configuration examples for CleanAir

This example shows how to enable CleanAir on the 2.4-GHz band and an AP operating in the channel:

```
Device#configure terminal
Device(config)#ap dot11 24ghz cleanair
Device(config)#exit
Device#ap name TAP1 dot11 24ghz cleanair
Device#end
```

This example shows how to enable an EDRRM CleanAir event in the 2.4-GHz band and configure high sensitivity to non-Wi–Fi interference:

```
Device#configure terminal
Device(config)#ap dot11 24ghz rrm channel cleanair-event
Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high
Device(config)#end
```

Verify CleanAir parameters

You can verify CleanAir parameters using these commands:

Table 38: Commands for verifying CleanAir

Command Name	Description
show ap dot11 24ghz cleanair device type all	Displays all the CleanAir interferers for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-discovery	Displays CleanAir interferers of type BT Discovery for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-link	Displays CleanAir interferers of type BT Link for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type canopy	Displays CleanAir interferers of type Canopy for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type cont-tx	Displays CleanAir interferers of type Continuous transmitter for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type dect-like	Displays CleanAir interferers of type Digital Enhanced Cordless Telecommunications (DECT) Like for the 2.4-GHz band.

Command Name	Description
show ap dot11 24ghz cleanair device type fh	Displays CleanAir interferers of type 802.11FH for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type inv	Displays CleanAir interferers of type Wi-Fi Inverted for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type jammer	Displays CleanAir interferers of type Jammer for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type mw-oven	Displays CleanAir interferers of type MW Oven for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type nonstd	Displays CleanAir interferers of type Wi-Fi inverted channel for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type superag	Displays CleanAir interferers of type SuperAG for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type tdd-tx	Displays CleanAir interferers of type TDD Transmit for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type video	Displays CleanAir interferers of type Video Camera for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type wimax-fixed	Displays CleanAir interferers of type WiMax Fixed for the 2.4-GHz band.

Configure air quality traps (CLI)

Set up SNMP traps so that you receive alerts when the air quality falls below a specified threshold.

Procedure

Step 1 Configure the trap **ciscoLwappSiAqLowSeverityHigh** to generate an alert when air quality goes beyond a given threshold.

On the controller GUI, navigate to **Configuration** > **Radio Configurations** > **Configuration** > **CleanAir** > **Trap configuration**.

The trap **ciscoLwappSiAqLowSeverityHigh** (OID: 1.3.6.1.4.1.9.9.513.1.6.1.0.5) is used to monitor air quality and generate alerts when the air quality index exceeds a specified threshold.

Step 2 Send the trap from the controller CLI. From the global configuration mode, enable SNMP traps for air quality index (AQI).

Example:

```
Device# snmp-server enable traps wireless SI
```

SNMP traps are configured, and the device sends alerts when air quality index thresholds are exceeded.

What to do next

Monitor SNMP alerts to ensure notifications are received as expected.

Monitor with CleanAir device clusters

A CleanAir device cluster is a spectrum monitoring feature that

- merges detections of the same interference device from multiple sensors into a single cluster,
- assigns and maintains a unique cluster ID for each device, and
- preserves device history by keeping cluster IDs active during temporary lapses in detection.

How CleanAir handles interference device monitoring

Some devices conserve power by limiting the transmit time until actually needed, which results in the spectrum sensor to stop detecting the device temporarily. This device is then correctly marked as down and removed from the spectrum database.

In cases when all the interferer detections for a specific device are reported, the cluster ID is kept alive for an extended period of time to prevent possible device-detection bouncing. If the same device is detected again, it is merged with the original cluster ID and the device-detection history is preserved.

: Bluetooth headset power management

Some Bluetooth headsets operate on battery power and employ methods to reduce power consumption, such as turning off the transmitter when not needed. Such devices can appear to come and go from the classification. To manage these devices, CleanAir keeps the cluster IDs for longer and they are remerged into a single record upon detection. This process smoothens the user records and accurately represents the device history.

CleanAir FAQs

Q. Multiple access points detect the same interference device. However, the device shows them as separate clusters or different suspected devices clustered together. Why does this happen?

A. Access points must be RF neighbors for the device to consider merging the devices that are detected by these access points. An access point takes time to establish neighbor relationships. A few minutes after the device reboots or after there is a change in the RF group, and similar events, clustering will not be very accurate.

Q. How do I view neighbor access points?

A. To view neighbor access points, use the **show ap** *ap_name* **auto-rf dot11** { **24ghz** | **5ghz** } command.

This example shows how to display the neighbor access points:

```
Device#show ap name AS-5508-5-AP3 auto-rf dot11 24ghz

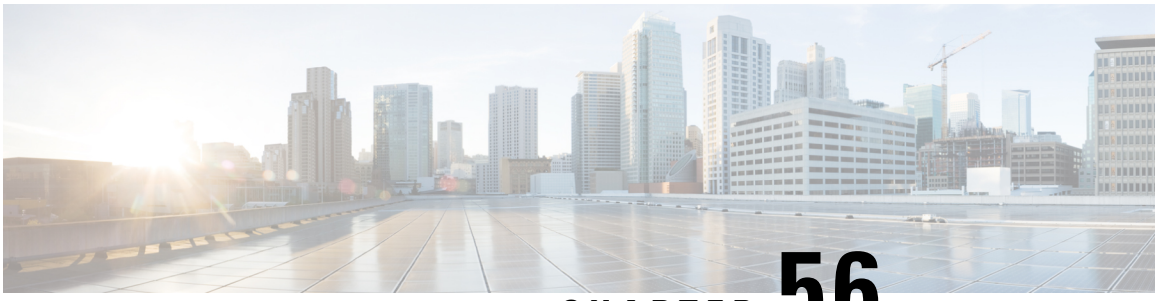
<snippet>
Nearby APs
  AP 0C85.259E.C350 slot 0                        :  -12 dBm on   1 (10.10.0.5)
  AP 0C85.25AB.CCA0 slot 0                        :  -24 dBm on   6 (10.10.0.5)
  AP 0C85.25C7.B7A0 slot 0                        :  -26 dBm on  11 (10.10.0.5)
```

```
  AP 0C85.25DE.2C10 slot 0                          :  -24 dBm on   6 (10.10.0.5)
  AP 0C85.25DE.C8E0 slot 0                          :  -14 dBm on  11 (10.10.0.5)
  AP 0C85.25DF.3280 slot 0                          :  -31 dBm on   6 (10.10.0.5)
  AP 0CD9.96BA.5600 slot 0                          :  -44 dBm on   6 (10.0.0.2)
  AP 24B6.5734.C570 slot 0                          :  -48 dBm on  11 (10.0.0.2)
<snippet>
```

Q. What are the AP debug commands available for CleanAir?

A. The AP debug commands for CleanAir are:

-
-



CHAPTER 56

Spectrum Intelligence

- Spectrum Intelligence, on page 617

Spectrum Intelligence

A spectrum intelligence feature is a radio analysis tool that

- scans 2.4-GHz and 5-GHz bands for non-Wi-Fi radio interference
- detects and classifies interference types such as microwave, continuous wave (like video bridge and baby monitor), Wi-Fi, and frequency hopping (Bluetooth and frequency-hopping spread spectrum (FHSS) cordless phone), and
- provides detailed radio utilization, noise, air quality, and interference reports for compatible Cisco APs.

For information about APs that support this feature, see the relevant documentation. https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

Note You must enable the Spectrum Intelligence feature on Cisco Aironet 1832 and 1852 series APs to view radio details such as noise, air quality, interference, and radio utilization in Cisco Catalyst Center Assurance AP health.

Restrictions

- SI APs report only a single interference type in Local mode.
- SI does not support high availability for air quality or interference reports. High availability is not supported because the interference report or device report will not be copied to standby after switchover. The AP is expected to send the report again if the interfering device is still present.
- Spectrum Intelligence detects only three types of devices:
  - Microwave
  - continuous wave (such as video recorder, baby monitor)
  - SI-FHSS (Bluetooth and frequency hopping Digital European Cordless Telecommunications [DECT] phones).

Configure spectrum intelligence (CLI)

Enable network administrators to monitor and analyze the radio frequency (RF) environment by configuring the Spectrum Intelligence (SI) feature on Cisco APs.

Follow the procedure given below to configure spectrum intelligence:

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure the 2.4-GHz or 5-GHz spectrum intelligence feature on the 802.11a or 802.11b network.

Example:

```
Device(config)# ap dot11 {24ghz | 5ghz} SI
```

Add **no** form of the command to disable SI on the 802.11a or 802.11b network.

Verify spectrum intelligence information

Use these commands to verify spectrum intelligence information:

To display the SI information for a 2.4-GHz or 5-GHz band, use this command:

```
Device# show ap dot11 24ghz SI config

SI Solution..................................... : Enabled
Interference Device Settings:
SI_FHSS............................... : Enabled
Interference Device Types Triggering Alarms:
SI_FHSS............................... : Disabled
```

To display SI interferers of type continuous transmitter for a 2.4-GHz band, use this command:

```
Device# show ap dot11 24ghz SI device type cont_tx

DC    = Duty Cycle (%)
ISI = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
AP type = CA, clean air, SI spectrum intelligence
No  ClusterID       DevID  Type     AP Type AP Name                ISI  RSSI  DC   Channel
--- --------------- ------ -------- ------- ------------------ ---- ----- ----  ---  ------
xx:xx:xx:xx  0014  BT     CA    myAP1              --   -69 00  133
xx:xx:xx:xx  0014  BT     SI    myAP1              --   -69 00  133
```

To display 802.11a interference devices information for the given AP for 5-GHz, use this command:

```
Device# show ap dot11 5ghz SI device type ap

DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
```

```
DevID = Device ID
AP type = CA, clean air, SI spectrum intelligence
No  ClusterID/BSSID    DevID  Type    AP Type AP Name                ISI  RSSI  DC   Channel
--- ------------------ ------ ------- ------  ---------------------- ---- ----- ---- ----------
```

To display all Cisco CleanAir interferers for a 2.4-GHz band, use this command:

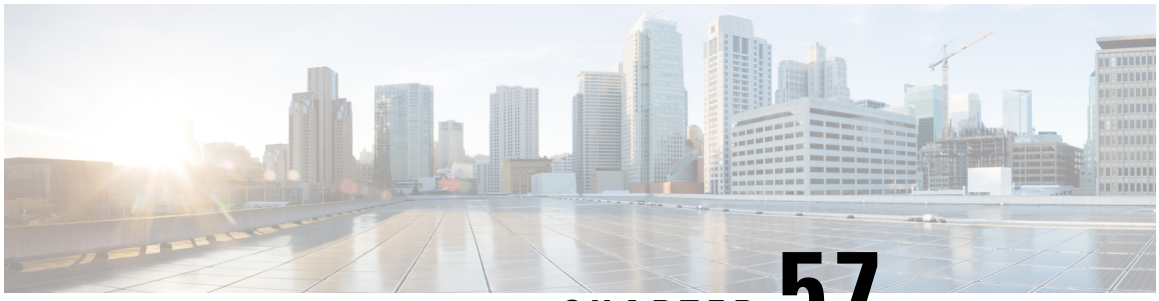
```
Device# show ap dot11 24ghz cleanair device type all
```



PART IX

WLAN

- WLANs, on page 623
- Network Access Server Identifier, on page 639
- DHCP for WLANs, on page 645
- WLAN Security, on page 647
- Peer-to-Peer Client Support, on page 651
- 802.11r BSS Fast Transition, on page 653
- Assisted Roaming, on page 663
- 802.11v, on page 667
- 802.11w, on page 671



CHAPTER 57

WLANs

- Information About WLANs, on page 623
- Prerequisites for WLANs, on page 626
- Restrictions for WLANs, on page 626
- How to Configure WLANs, on page 628
- WLAN properties verification (CLI), on page 637

Information About WLANs

This feature enables you to control WLANs for lightweight access points. Each WLAN has a separate WLAN ID, a separate profile name, and a WLAN SSID. All access points can advertise up to 16 WLANs. However, you can create up to 4096 WLANs and then selectively advertise these WLANs (using profiles and tags) to different APs for better manageability.

You can configure WLANs with different SSIDs or with the same SSID. An SSID identifies the specific wireless network that you want the controller to access.

Band selection

A band selection is a wireless network feature that

- facilitates the movement of dual-band client radios from congested frequency bands
- enables clients to connect to less congested 5-GHz access points for improved network performance, and
- reduces interference from other devices by optimizing which frequency band clients use.

The 2.4-GHz band is often congested. Clients using this band may experience interference from Bluetooth devices, microwave ovens, cordless phones, and co-channel interference from other access points due to the 802.11b/g channel limit of 3 nonoverlapping channels. By configuring band selection on devices, administrators can steer dual-band clients toward the 5-GHz band, improving overall network performance.

Off-Channel Scanning Deferral

A lightweight access point, in normal operational conditions, periodically goes off-channel and scans another channel. This is in order to perform RRM operations such as the following:

- Transmitting and receiving Neighbor Discovery Protocol (NDP) packets with other APs.
- Detecting rogue APs and clients.
- Measuring noise and interference.

During the off-channel period, which normally is about 70 milliseconds, the AP is unable to transmit or receive data on its serving channel. Therefore, there is a slight impact on its performance and some client transmissions might be dropped.

While the AP is sending and receiving important data, it is possible to configure off-channel scanning deferral so that the AP does not go off-channel and its normal operation is not impacted. You can configure off-channel scanning deferral on a per-WLAN basis, per WMM UP class basis, with a specified time threshold in milliseconds. If the AP sends or receives, on a particular WLAN, a data frame marked with the given UP class within the specified threshold, the AP defers its next RRM off-channel scan. For example, by default, off-channel scanning deferral is enabled for UP classes 4, 5, and 6, with a time threshold of 100 milliseconds. Therefore, when RRM is about to perform an off-channel scan, a data frame marked with UP 4, 5, or 6 is received within the last 100 milliseconds, RRM defers going off-channel. The AP radio does not go off-channel when a voice call sending and receiving audio samples are marked as UP class 6 for every active 20 milliseconds.

Off-channel scanning deferral does come with a tradeoff. Off-channel scanning can impact throughput by 2 percent or more, depending on the configuration, traffic patterns, and so on. Throughput can be slightly improved if you enable off-channel scanning deferral for all traffic classes and increase the time threshold. However, by not going off-channel, RRM can fail to identify AP neighbors and rogues, resulting in negative impact to security, DCA, TPC, and 802.11k messages.

DTIM period

A Delivery Traffic Indication Map (DTIM) period is a 802.11 network timing mechanism that

- allows power-saving clients to wake up at the appropriate time if they are expecting broadcast or multicast data
- determines when access points transmit buffered broadcast and multicast frames after beacon broadcasts, and
- coincides with the DTIM broadcast interval.

DTIM period characteristics

In the 802.11 networks, lightweight access points broadcast a beacon at regular intervals, which coincides with the Delivery Traffic Indication Map (DTIM). After the access point broadcasts the beacon, it transmits any buffered broadcast and multicast frames based on the value set for the DTIM period.

Typical DTIM values and their transmission patterns:

- DTIM value 1: Transmits broadcast and multicast frames after every beacon
- DTIM value 2: Transmits broadcast and multicast frames after every other beacon

For instance, if the beacon period of the 802.11 network is 100 ms and the DTIM value is set to 1, the access point transmits buffered broadcast and multicast frames for 10 times every second. If the beacon period is 100 ms and the DTIM value is set to 2, the access point transmits buffered broadcast and multicast frames

for 5 times every second. Either of these settings are suitable for applications, including Voice Over IP (VoIP), that expect frequent broadcast and multicast frames.

However, the DTIM value can be set as high as 255 (to transmit broadcast and multicast frames after every 255th beacon). The only recommended DTIM values are 1 and 2; higher DTIM values will likely cause communications problems.



Note A beacon period, which is specified in milliseconds on the device, is converted internally by the software to 802.11 Time Units (TUs), where 1 TU = 1.024 milliseconds. Depending on the AP model, the actual beacon period may vary slightly; for example, a beacon period of 100 ms may in practice equate to 104.448 ms.

Session Timeouts

You can configure a WLAN with a session timeout. The session timeout is the maximum time for a client session to remain active before requiring reauthorization.

If a WLAN is configured with Layer 2 security, for example WPA2-PSK, and a Layer 3 authentication is also configured, the WLAN session timeout value is overridden with the 802.1X reauthentication timeout value. If APF reauthentication timeout value is greater than 65535, the WLAN session timeout is by default set to 65535; else, the configured 802.1X reauthentication timeout value is applied as the WLAN session timeout.

This section contains the following subsections:

Cisco Client Extensions

The Cisco Client Extensions (CCX) software is licensed to manufacturers and vendors of third-party client devices. The CCX code resident on these clients enables them to communicate wirelessly with Cisco access points and to support Cisco features that other client devices do not, including those features that are related to increased security, enhanced performance, fast roaming, and power management.

The software supports CCX versions 1 through 5, which enables devices and their access points to communicate wirelessly with third-party client devices that support CCX. CCX support is enabled automatically for every WLAN on the device and cannot be disabled. However, you can configure Aironet information elements (IEs).

If Aironet IE support is enabled, the access point sends an Aironet IE 0x85 (which contains the access point name, load, number of associated clients, and so on) in the beacon and probe responses of this WLAN, and the device sends Aironet IEs 0x85 and 0x95 (which contains the management IP address of the device and the IP address of the access point) in the reassociation response if it receives Aironet IE 0x85 in the reassociation request.

Peer-to-peer blocking

Peer-to-peer blocking is a WLAN security feature that

- is applied to individual WLANs, where each client inherits the peer-to-peer blocking setting of the WLAN to which it is associated
- enables you to have more control over how traffic is directed by allowing you to choose to have traffic bridged locally within the device, dropped by the device, or forwarded to the upstream VLAN, and

- is supported for clients that are associated with local and central switching WLANs.

Note Peer-to-peer blocking feature is VLAN-based. WLANs using the same VLAN has an impact, if Peer-to-peer blocking feature is enabled.

Diagnostic channels

A diagnostic channel is a troubleshooting feature that

- enables testing of client and AP communication to identify network difficulties
- allows corrective measures to be taken to make the client operational on the network, and
- provides GUI or CLI configuration options for enabling diagnostic tests.

Configuration options

You can use the device GUI or CLI to enable the diagnostic channel, and you can use the device **diag-channel** CLI to run the diagnostic tests.

Note We recommend that you enable the diagnostic channel feature only for non-anchored SSIDs that use the management interface. CCX Diagnostic feature has been tested only with clients having Cisco ADU card

Prerequisites for WLANs

You can associate up to 16 WLANs with each policy tag.

We recommend that you assign one set of VLANs for WLANs and a different set of VLANs for management interfaces to ensure that devices properly route VLAN traffic.

Restrictions for WLANs

This reference provides restrictions and limitations that apply when configuring and managing WLANs on wireless controllers.

- Do not configure PSK as this configuration is not supported and impacts client join flow.
- Ensure that TKIP or AES ciphers are enabled with WPA1 configuration, else ISSU may break during upgrade process.
- When you change the WLAN profile name, then FlexConnect APs (using AP-specific VLAN mapping) will become WLAN-specific. If FlexConnect Groups are configured, the VLAN mapping will become Group-specific.

- Do not enable IEEE 802.1X Fast Transition on Flex Local Authentication enabled WLAN, as client association is not supported with Fast Transition 802.1X key management.
- Peer-to-peer blocking does not apply to multicast traffic.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- The WLAN name and SSID can have up to 32 characters.
- WLAN and SSID names support only the following ASCII characters:
  - Numerals: 48 through 57 hex (0 to 9)
  - Alphabets (uppercase): 65 through 90 hex (A to Z)
  - Alphabets (lowercase): 97 through 122 hex (a to z)
  - ASCII space: 20 hex
  - Printable special characters: 21 through 2F, 3A through 40, and 5B through 60 hex, that is: ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
- WLAN name cannot be a keyword; for example, if you try to create a WLAN with the name as 's' by entering the **wlan s** command, it results in shutting down all WLANs because 's' is used as a keyword for shutdown.
- You cannot map a WLAN to VLAN 0. Similarly, you cannot map a WLAN to VLANs 1002 to 1006.
- Dual stack clients with a static-IPv4 address is not supported.
- In a dual-stack with IPv4 and IPv6 configured in the Cisco 9800 controller, if an AP tries to join controller with IPv6 tunnel before its IPv4 tunnel gets cleaned, you would see a traceback and AP join will fail.
- When creating a WLAN with the same SSID, you must create a unique profile name for each WLAN.
- When multiple WLANs with the same SSID get assigned to the same AP radio, you must have a unique Layer 2 security policy so that clients can safely select between them.
- If the newly configured SSID is on a 5-GHz DFS channel, beaconing does not start immediately.
- RADIUS server overwrite is not configured on a per WLAN basis, but rather on a per AAA server group basis.
- Downloadable ACL (DACL) is supported only on the central switching mode. It is not supported for Flex Local switching or on the Cisco Embedded Wireless Controller.



Caution Some clients might not be able to connect to WLANs properly if they detect the same SSID with multiple security policies. Use this WLAN feature with care.

How to Configure WLANs

Create WLANs (GUI)

Create wireless local area networks (WLANs) to enable network connectivity and manage wireless access configurations through the GUI interface.

Use the GUI interface to create and configure WLANs when you need to establish wireless network connectivity. This method provides a visual interface for WLAN configuration and management.

Procedure

Step 1 In the **Configuration** > **Tags & Profiles** > **WLANs** page, click **Add**.

The **Add WLAN** window is displayed.

Step 2 Under the **General** tab and **Profile Name** field, enter the name of the WLAN. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 3 Click **Save & Apply to Device**.

The WLAN is created and applied to the device. The new WLAN appears in the WLANs list and is ready for use.

Create WLANs (CLI)

Create wireless local area networks (WLANs) to provide wireless network access for clients.

You can create SSID using GUI or CLI. However, we recommend that you use CLI to create SSID. By default, the WLAN is disabled after creation.

Follow these steps to create WLANs using CLI commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Specify the WLAN name and ID.

Example:

```
Device(config)# wlan profile-name wlan-id [ssid]
```

Example:

```
Device(config)# wlan profile-name wlan-id [ssid] mywlan 34 mywlan-SSID
```

- For the *profile-name*, enter the profile name. The range is from 1 to 32 alphanumeric characters.
- For the *wlan-id*, enter the WLAN ID. The range is from 1 to 512.
- For the *ssid*, enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

Note
- You can create SSID using GUI or CLI. However, we recommend that you use CLI to create SSID.
- By default, the WLAN is disabled.

Step 3 Return to privileged EXEC mode.

Example:

```
Device# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The WLAN is created with the specified profile name, WLAN ID, and SSID. The WLAN is disabled by default and must be enabled separately for use.

Delete WLANs (GUI)

Remove unwanted or obsolete WLANs from your network configuration to maintain optimal performance and security.

Use this procedure when you need to remove WLANs that are no longer required or need to be replaced with new configurations. This task can be performed on single or multiple WLANs simultaneously.

Procedure

Step 1 In the **Configuration** > **Tags & Profiles** > **WLANs** page, check the check box adjacent to the WLAN you want to delete.

To delete multiple WLANs, select multiple WLANs check boxes.

Step 2 Click **Delete**.

Step 3 Click **Yes** on the confirmation window to delete the WLAN.

The selected WLAN(s) are permanently removed from the system and will no longer appear in the WLANs list.

Delete WLANs (CLI)

Delete WLAN profiles that are no longer needed or require reconfiguration in the wireless network environment.

Use this procedure when you need to remove WLAN profiles from the wireless controller configuration. Deleting a WLAN removes the profile and associated settings from the system.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Delete the WLAN.

Example:

```
Device(config)# no wlan wlan-profile-name wlan-id  [ssid]
Device(config)# no wlan test2
```

The arguments are:

- The *wlan-name* is the WLAN profile name.
- The *wlan-id* is the WLAN ID.
- The *ssid* is the WLAN SSID name configured for the WLAN.

Step 3 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

The WLAN profile is deleted from the wireless controller configuration. If the WLAN was associated with an AP group, it is automatically removed from the group and the AP's radio interface.

WLAN search methods (CLI)

To verify the list of all WLANs configured on the controller, use this command:

```
Device# show wlan summary
Number of WLANs: 4

WLAN Profile Name                     SSID                           VLAN Status
-------------------------------------------------------------------------------
1    test1                            test1-SSID                     137  UP
3    test2                            test2-SSID                     136  UP
2    test3                            test3-SSID                     1    UP
45   test4                            test4-SSID                     1    DOWN
```

To use wild cards and search for WLANs, use this **show** command:

```
Device# show wlan summary | include test-wlan-ssid
1    test-wlan                       test-wlan-ssid                   137   UP
```

Enable WLANs (GUI)

Enable wireless local area networks (WLANs) to allow wireless client connections and network access through the graphical user interface.

Use this task when you need to activate WLANs that have been configured but are currently disabled, allowing wireless devices to connect to your network.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **WLANs** page, click the WLAN name.

Step 3 In the **Edit WLAN** window, toggle the **Status** button to **ENABLED**.

Step 4 Click **Update & Apply to Device**.

The WLAN is ENABLED and wireless clients can now connect to the network through this WLAN.

Enable WLANs (CLI)

Enable previously configured WLANs to make them operational and available for client connections.

WLANs must be explicitly enabled after configuration to allow wireless clients to connect. This procedure activates a specific WLAN profile.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Enter WLAN configuration submode.

Example:

```
Device(config)# WLAN profile-name
```

Example:

```
Device(config)# WLAN test4
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

The WLAN is now enabled and operational, allowing wireless clients to connect to the network.

Disable WLANs (GUI)

Disable WLANs to prevent wireless network access when maintenance is required or to temporarily suspend wireless services.

Use this procedure when you need to temporarily disable wireless networks without deleting the WLAN configuration. The WLAN settings remain intact and can be re-enabled later.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 In the **WLANs** window, click the WLAN name.

Step 3 In the **Edit WLAN** window, set the **Status** toggle button as **DISABLED**.

Step 4 Click **Update & Apply to Device**.

The WLAN is DISABLED and wireless clients can no longer connect to this network. The WLAN configuration is preserved and can be re-enabled when needed.

Disable WLANs (CLI)

Disable specific WLANs to prevent wireless client access when needed for maintenance or security purposes.

You may need to disable WLANs temporarily for maintenance, troubleshooting, or security reasons. This procedure shows how to disable a WLAN using the CLI and verify the change.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device(config)# wlan profile-name
```

Example:

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Disable the WLAN.

Example:

```
Device(config-wlan)# shutdown
```

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Step 5 Display the list of all WLANs configured on the device.

Example:

```
Device# show wlan summary
```

You can search for the WLAN in the output to verify it has been disabled.

The specified WLAN is now disabled and will not accept wireless client connections until re-enabled.

Configure general WLAN properties (CLI)

Configure media stream, broadcast SSID, and radio properties for a WLAN.

You can configure these properties: Media stream, Broadcast SSID, and Radio settings for optimal WLAN performance and visibility.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device(config)# wlan profile-name
```

Example:

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Disable the WLAN.

Example:

```
Device(config-wlan)# shutdown
```

Step 4 Broadcast the SSID for this WLAN.

Example:

```
Device(config-wlan)# broadcast-SSID
```

Step 5 Configure the WLAN radio policy for dot11 radios.

Example:

```
Device(config-wlan)# dot11bg 11g
```

Step 6 Enable multicast VLANs on this WLAN.

Example:

```
Device(config-WLAN)# media-stream multicast-direct
```

Step 7 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 8 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

The WLAN is now configured with the specified general properties including media stream, broadcast SSID, and radio settings.

Configure advanced WLAN properties (CLI)

Configure advanced WLAN properties to customize wireless network behavior and security settings.

Advanced WLAN properties provide additional configuration options for wireless networks, including coverage hole detection, client association limits, access control, and peer-to-peer blocking settings. These configurations help optimize wireless performance and security based on specific network requirements.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device(config)# wlan profile-name
```

Example:

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Enable coverage hole detection for this WLAN.

Example:

```
Device(config-wlan)# chd
```

Step 4 Enable support for Aironet IEs for this WLAN.

Example:

```
Device(config-wlan)# ccx aironet-iesupport
```

Step 5 Set the maximum number of clients that can be configured on a WLAN. Configure limits for clients per WLAN, clients per AP, or clients per AP radio.

Example:

```
Device(config-wlan)# client association limit {clients-per-WLAN | ap clients-per-AP-per-WLAN
 | radio clients-per-AP-radio-per-WLAN}
```

Example:

```
Device(config-wlan)# client association limit AP 400
```

Step 6 Configure the IPv4 WLAN web ACL.

Example:

```
Device(config-wlan)# ip access-group web ACL-name
```

Example:

```
Device(config-wlan)# ip access-group web test-ACL-name
```

The *ACL-name* specifies the user-defined IPv4 ACL name.

Step 7 Configure peer-to-peer blocking parameters.

Example:

```
Device(config-wlan)# peer-blocking [drop | forward-upstream]
```

Example:

```
Device(config-WLAN)# peer-blocking drop
```

The keywords are:

- **drop**: Enables peer-to-peer blocking on the drop action.
- **forward-upstream**: No action is taken and forwards packets to the upstream.

Step 8 Set the channel scan defer priority and defer time.

Example:

```
Device(config-WLAN)# channel-scan {defer-priority {0-7} | defer-time {0-6000}}
```

Example:

```
Device(config-WLAN)# channel-scan defer-priority 6
```

The arguments are:

- **defer-priority**: Specifies the priority markings for packets that can defer off-channel scanning. The range is from 0 to 7. The default is 3.
- **defer-time**: Deferral time in milliseconds. The range is from 0 to 6000. The default is 100.

Step 9 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

The advanced WLAN properties are now configured with the specified settings for coverage hole detection, client limits, access control, peer-to-peer blocking,broadcast filtering, AAA override, session timeout, exclusion list management, and diagnostic channel support.

Configure advanced WLAN properties (GUI)

Configure advanced WLAN properties to enable enhanced wireless network features including coverage optimization, client connection limits, BSS transition support, and 802.11ax capabilities.

Advanced WLAN properties provide additional configuration options to optimize wireless network performance and enable specific features for client management and network diagnostics.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.

Step 2 In the **Wireless Networks** window, click **Add**.

Step 3 Under the **Advanced** tab, check the **Coverage Hole Detection** check box.

Step 4 Check the **Aironet IE** check box to enable Aironet IE on the WLAN.

Step 5 Check the **Diagnostic Channel** check box to enable diagnostic channel on the WLAN.

Step 6 From the **P2P Blocking Action** drop-down list, choose the required value.

Step 7 Set the **Multicast Buffer** toggle button as enabled or diabled.

Step 8 Check the **Media Stream Multicast-Direct** check box to enable the feature.

Step 9 In the **Max Client Connections** section, specify the maximum number of client connections for the following:

- In the **Per WLAN** field, enter a value. The valid range is between 0 and 10000.
- In the **Per AP Per WLAN** field, enter a value. The valid range is between 0 and 400.
- In the **Per AP Radio Per WLAN** field, enter a value. The valid range is between 0 and 200.

Step 10 In the **11v BSS Transition Support** section, perform the following configuration tasks:

a) Check the BSS Transition check box to enable 802.11v BSS Transition support.
b) In the **Disassociation Imminent** field, enter a value. The valid range is between 0 and 3000.
c) In the **Optimized Roaming Disassociation Timer** field, enter a value. The valid range is between 0 and 40.
d) Select the check box to enable:

- BSS Max Idle Service
- BSS Max Idle Protected
- Disassociation Imminent Service
- Directed Multicast Service
- Universal Admin
- Load Balance
- Band Select

- IP Source Guard

Step 11 From the **WMM Policy** drop-down list, choose the policy as Allowed, Disabled, or Required. By default, the WMM policy is Allowed.

Step 12 In the **Off Channel Scanning Defer** section, choose the appropriate **Defer Priority** values and then specify the required Scan Defer Time value in milliseconds.

Step 13 In the **Assisted Roaming (11k)** section, choose the appropriate status for:

- Prediction Optimization
- Neighbor List
- Dual-Band Neighbor List

Step 14 In the **DTIM Period (in beacon intervals)** section, specify a value for 802.11a/n and 802.11b/g/n radios. The valid range is from 1 to 255.

Step 15 Click **Save & Apply to Device**.

The WLAN profile is updated with advanced properties, and features such as coverage optimization, client connection limits, enhanced roaming, and diagnostics are enabled for improved network performance.

WLAN properties verification (CLI)

To verify the WLAN properties based on the WLAN ID, use this command:

```
Device# show wlan ID wlan-id
```

To verify the WLAN properties based on the WLAN name, use this command:

```
Device# show wlan name wlan-name
```

To verify the WLAN properties of all the configured WLANs, use this command:

```
Device# show wlan all
```

To verify the summary of all WLANs, use this command:

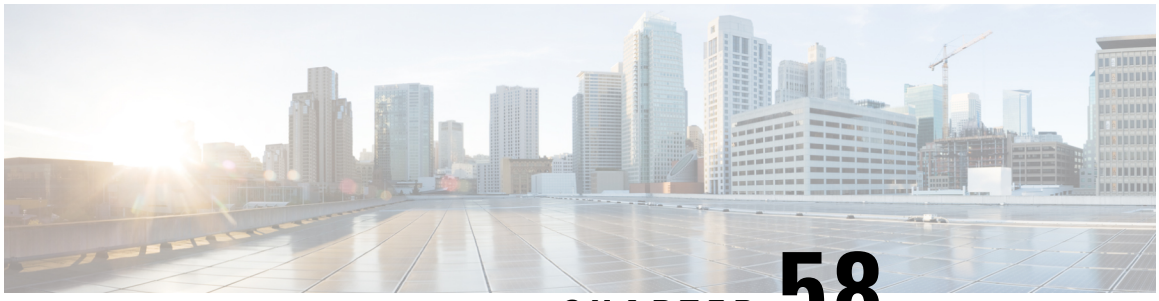
```
Device# show wlan summary
```

To verify the running configuration of a WLAN based on the WLAN name, use this command:

```
Device# show running-config wlan wlan-name
```

To verify the running configuration of all WLANs, use this command:

```
Device# show running-config wlan
```



CHAPTER 58

Network Access Server Identifier

- Network Access Server Identifier (NAS-ID), on page 639

Network Access Server Identifier (NAS-ID)

A network access server identifier (NAS-ID) is a RADIUS attribute that

- identifies the source device, WLAN profile, VLAN interface, or AP group initiating a RADIUS access request
- enables the RADIUS server to apply appropriate authentication and policy selection, and
- supports customized policy enforcement based on user groups or device types.

The NAS-ID is sent to the RADIUS server by the embedded wireless controller through an authentication request to classify users to different groups. This enables the RADIUS server to send a customized authentication response.

Note The acct-session-id is sent with the RADIUS access request only if accounting is enabled on the policy profile.

If you configure a NAS-ID for a WLAN profile, it overrides the NAS-ID that is configured for the VLAN interface.

These options can be configured for a NAS ID:

- sys-name (System Name)
- sys-ip (System IP Address) and sys-mac (System MAC Address)
- ap-ip (AP's IP address) and ap-name (AP's Name)
- ap-mac (AP's MAC Address)
- ap-eth-mac (AP's Ethernet MAC Address)
- ap-policy-tag (AP's policy tag name)
- ap-site-tag (AP's site tag name)
- ssid (SSID Name) and ap-location (AP's Location)

Create a NAS ID policy (GUI)

Configure a NAS ID policy that defines how network access server identifiers are applied within your wireless AAA policy using the GUI.

Creating a NAS ID policy ensures correct identification and handling of wireless authentication requests. This is typically performed when setting up or modifying wireless AAA policies.

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless AAA Policy**.

Step 2 On the **Wireless AAA Policy** page, click the name of the **Policy** or click **Add** to create a new one.

Step 3 In the **Add/Edit Wireless AAA Policy** window that is displayed, enter the name of the policy in the **Policy Name** field.

Step 4 Select from one of the NAS ID options from the **Option 1** drop-down list.

Step 5 Select from one of the NAS ID options from the **Option 2** drop-down list.

Step 6 Select from one of the NAS ID options from the **Option 3** drop-down list.

Step 7 Save the configuration.

Create a NAS ID policy (CLI)

Create and configure a NAS ID policy for wireless RADIUS authentication and accounting using commands.

Follow the procedure given below to create NAS ID policy:

Before you begin

- A NAS ID can combine up to a maximum of three options.
- The maximum length of the NAS ID attribute is 253. Before adding a new attribute, the system checks the attribute buffer. If there is not enough space, the system ignores the new attribute.
- By default, a wireless aaa policy (default-aaa-policy) is created with the default configuration (sys-name). You can update this policy with various NAS ID options. However, the default-aaa-policy cannot be deleted.
- If you do not configure a NAS ID, the system uses the default sys-name as the NAS ID for all wireless-specific RADIUS packets (authentication and accounting) from the embedded wireless controller.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a new AAA policy.

Example:

```
Device(config)# wireless aaa policy policy-name
```

Step 3 Configure NAS ID for option1.

Example:

```
Device(config-aaa-policy)# nas-id option1 sys-name
```

Step 4 Configure NAS ID for option2.

Example:

```
Device(config-aaa-policy)# nas-id option2 sys-ip
```

Step 5 Configure NAS ID for option3.

Example:

```
Device(config-aaa-policy)# nas-id option3 sys-mac
```

Attach a policy to a tag (GUI)

Map a policy to a tag for consistent network device configuration using the GUI.

Attach a predefined policy to a tag to apply specific settings or profiles across selected devices.
Before you begin

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click **Add** and enter a name for the new policy, for example, test1, in the **General** tab.

Step 3 Click the **Advanced** tab, and under **AAA Policy**, from the **Policy Name** drop-down list, select the policy name that you had created in the **General** tab.

Step 4 Click **Apply to Device**.

Step 5 Choose **Configuration** > **Tags & Profiles** > **Tags** > **Policy**.

Step 6 Click **Add** to view the **Add Policy Tag** window.

Step 7 Enter a name and description for the policy tag.

Step 8 Click **Add** to map WLAN profile and policy profile.

Step 9 Choose the **WLAN Profile** to map with the appropriate **Policy Profile**, and click the tick icon. Click **Save & Apply to Device**.

Attach a policy to a tag (CLI)

Configure and attach a NAS ID policy to a wireless policy tag using commands.

Follow the procedure given below to attach a NAS ID policy to a tag:

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a WLAN policy profile.

Example:

```
Device(config)# wireless profile policy policy-name
```

Step 3 Configure a AAA policy profile.

Example:

```
Device(config-wireless-policy)# aaa-policy aaa-policy-name
```

Step 4 Return to the global configuration mode.

Example:

```
Device(config-wireless-policy)# exit
```

Step 5 Configure a wireless policy tag.

Example:

```
Device(config)# wireless tag policy policy-tag
```

Step 6 Map a WLAN profile to a policy profile.

Example:

```
Device(config)# wlan wlan1 policy policy-name
```

Note
You can also use the **ap-tag** option to configure a NAS ID for an AP group. This NAS ID overrides the NAS ID configured for a WLAN profile or VLAN interface.

Verify the NAS ID configuration

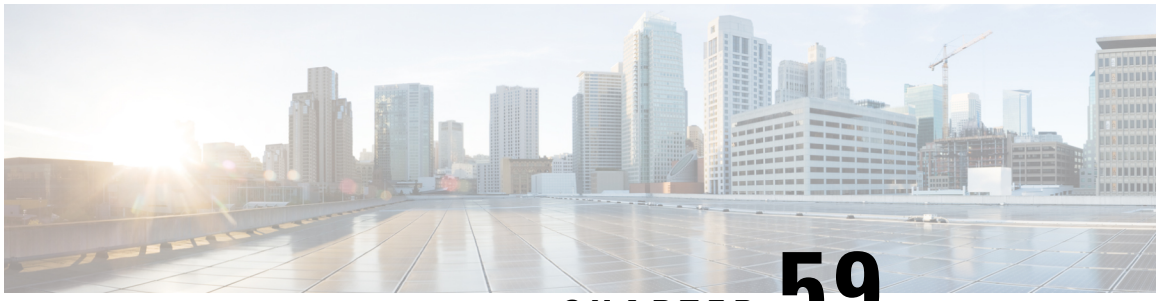
Use this **show** command to verify the NAS ID configuration:

```
Device# show wireless profile policy detailed test1

Policy Profile Name           : test1
Description                   :
Status                        : ENABLED
VLAN                          : 1
Client count                  : 0

:
:
AAA Policy Params
  AAA Override                : DISABLED
```

```
NAC                          : DISABLED
AAA Policy name              : test
```



CHAPTER 59

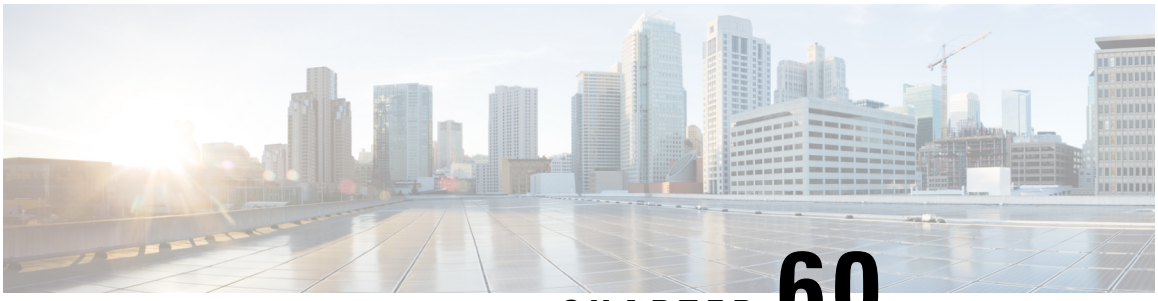
DHCP for WLANs

• DHCP for WLANs, on page 645

DHCP for WLANs

DHCP packets sent by the wireless clients are released in their respective VLANs as broadcast by the AP and relies on the fact that the network gateway of that VLAN forwards the requests to the DHCP server.

Note Internal DHCP server is not supported in EWC.



CHAPTER 60

WLAN Security

- AAA override, on page 647
- Prerequisites for Layer 2 Security, on page 647
- How to Configure WLAN Security, on page 648

AAA override

AAA override is a WLAN option that enables you to configure the WLAN for identity networking by applying VLAN tagging, Quality of Service (QoS), and Access Control Lists (ACLs) to individual clients based on the returned RADIUS attributes from the AAA server.

Prerequisites for Layer 2 Security

WLANs with the same SSID must have unique Layer 2 security policies so that clients can make a WLAN selection based on the information advertised in beacon and probe responses.

The available Layer 2 security policies are as follows:

- None (open WLAN)
- WPA+WPA2

Note

- Although WPA and WPA2 cannot be used by multiple WLANs with the same SSID, you can configure two WLANs with the same SSID with WPA/TKIP with PSK and Wi-Fi Protected Access (WPA)/Temporal Key Integrity Protocol (TKIP) with 802.1X, or with WPA/TKIP with 802.1X or WPA/AES with 802.1X.
- A WLAN configured with TKIP support will not be enabled on an RM3000AC module.

- Static WEP (not supported on Wave 2 APs)

How to Configure WLAN Security

Configure static WEP layer 2 security parameters (CLI)

Set up static WEP encryption and authentication for WLAN security.

Static WEP provides basic wireless security through shared encryption keys. This configuration is used when legacy devices require WEP compatibility, though WPA/WPA2 is recommended for better security.

Before you begin

You must have administrator privileges.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Static WEP layer 2 security is now configured on the WLAN with the specified authentication and encryption parameters.

Configure WPA + WPA2 layer 2 security parameters (CLI)

This task configures WPA and WPA2 layer 2 security settings to secure wireless networks using command-line interface commands.

Before you begin

You must have administrator privileges.

Follow these steps to configure WPA + WPA2 layer 2 security parameters using commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 **Example:**

```
Device(config-wlan)# security wpa {akm | wpa1 | wpa2}
```

```
Device(config-wlan)# security wpa
```

Step 3 Enables .

Example:

```
Device(config-wlan)# security wpa wpa1
```

Step 4 Specify the WPA1 cipher.

Example:

```
Device(config-wlan)# security wpa wpa1 ciphers [aes | tkip]
Device(config-wlan)# security wpa wpa1 ciphers aes
```

Choose one of the following encryption types:

- **aes**: Specifies WPA/AES support.
- **tkip**: Specifies WPA/TKIP support.

Step 5 Enable or disable Cisco Centralized Key Management, 802.1x, 802.1x with SHA256 key derivation type, Fast Transition, PSK or PSK with SHA256 key derivation type.**security WPA AKM** {**CCKM**| **dot1x** | **dot1x-sha256** | **ft** | **PSK** |**PSK-sha256**}

Example:

```
Device(config-wlan)# security wpa akm {cckm | dot1x | dot1x-sha256 | ft | psk | psk-sha256}
Device(config-wlan)# security wpa akm psk-sha256
```

Note

- You cannot enable 802.1x and PSK with SHA256 key derivation type simultaneously.
- When you configure Cisco Centralized Key Management SSID, you must enable the **ccx aironet-iesupport** for Cisco Centralized Key Management to work.
- WPA3 Enterprise dot1x-sha256 is supported only in local mode.

Step 6 Enable WPA2.

Example:

```
Device(config-wlan)# security wpa wpa2
```

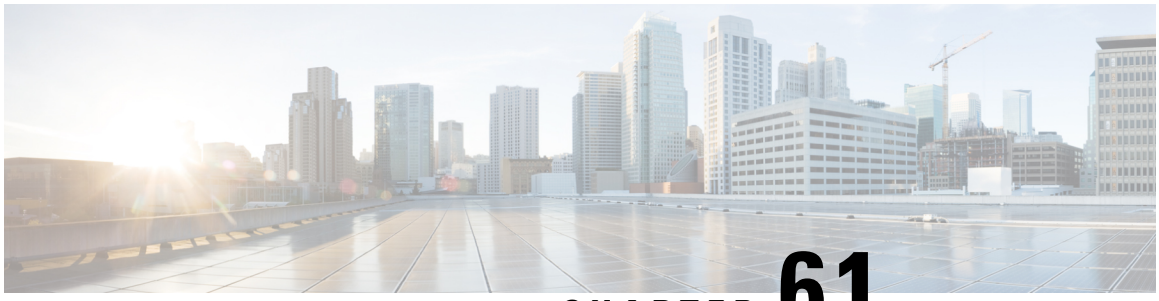
Step 7 Configure WPA2 cipher. **security wpa wpa2 ciphers aes**

Example:

```
Device(config-wlan)# security wpa wpa2
```

Example:

WPA and WPA2 layer 2 security parameters are configured for the WLAN. The wireless network is secured with the specified authentication and encryption settings.



CHAPTER 61

Peer-to-Peer Client Support

- Peer-to-peer client support, on page 651

Peer-to-peer client support

Peer-to-peer client support is a WLAN feature that

- provides granular control over how traffic is directed
- can be applied to individual WLANs with each client inheriting the peer-to-peer blocking setting of the WLAN to which it is associated
- allows traffic to be bridged locally within a device, dropped by a device, or forwarded to the upstream VLAN.

Peer-to-peer blocking support details

Peer-to-peer blocking is supported for clients that are associated with local and central switching WLANs.

These restrictions apply to peer-to-peer client support:

- Peer-to-peer blocking does not apply to multicast traffic.
- Peer-to-peer blocking is not enabled by default.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- FlexConnect central switching clients supports peer-to-peer upstream-forward. However, this is not supported in the FlexConnect local switching. This is treated as peer-to-peer drop and client packets are dropped.

  FlexConnect central switching clients supports peer-to-peer blocking for clients associated with different APs. However, for FlexConnect local switching, this solution targets only clients connected to the same AP. FlexConnect ACLs can be used as a workaround for this limitation.

Configure peer-to-peer client support

Enable peer-to-peer blocking policies to control communication between wireless clients on the same WLAN.

Peer-to-peer blocking allows you to control whether wireless clients can communicate directly with each other or must route traffic through the upstream network infrastructure. This feature is useful for security and traffic management in enterprise environments.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device(config)# wlan profile-name
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Configure peer-to-peer blocking parameters.

Example:

```
Device(config-wlan)# peer-blocking  | drop | forward-upstream
```

Example:

```
Device(config-wlan)# peer-blocking drop
```

The keywords are:

- **drop**: Enables peer-to-peer blocking on the drop action.
- **forward-upstream**: No action is taken and forwards packets to the upstream.

Step 4 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Step 5 Display the details of the selected WLAN.

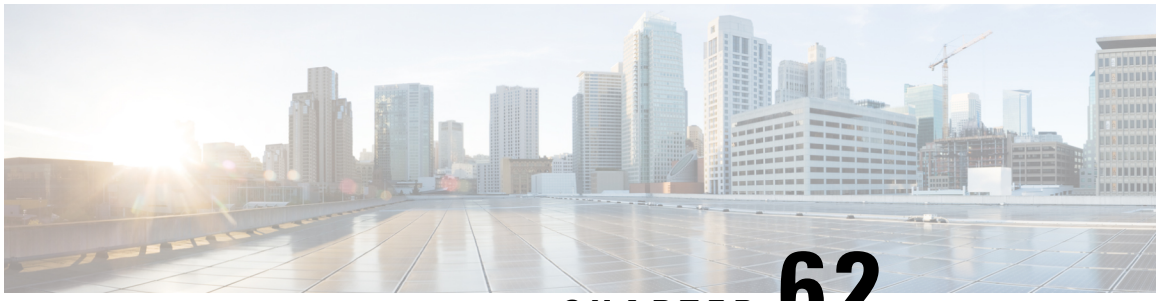
Example:

```
Device# show wlan id wlan-id
```

Example:

```
Device# show wlan id 12
```

Peer-to-peer client support is now configured on the WLAN with the specified blocking behavior for client-to-client communication.



CHAPTER 62

802.11r BSS Fast Transition

- 802.11r Fast Transition, on page 653

802.11r Fast Transition

802.11r Fast Transition is a roaming mechanism that

- enables the initial handshake with a new AP even before the corresponding client roams to the target access point
- allows a client and the access points to do the Pairwise Transient Key (PTK) calculation in advance, and
- applies PTK keys to the client and the access points after the client responds to the reassociation request or responds to the exchange with new target AP.

802.11r Fast Transition features

The FT key hierarchy is designed to allow clients to make fast BSS transitions between APs without requiring reauthentication at every AP. WLAN configuration contains a new Authenticated Key Management (AKM) type called Fast Transition (FT).

Client roaming

For a client to move from its current AP to a target AP using the FT protocols, message exchanges are performed using one of these methods:

- Over-the-Air: The client communicates directly with the target AP using IEEE 802.11 authentication with the FT authentication algorithm.
- Over-the-Distribution System (DS): The client communicates with the target AP through the current AP. The communication between the client and the target AP is carried in FT action frames between the client and the current AP and is then sent through the device.

Figure 10: Message exchanges when over–the–air client roaming is configured

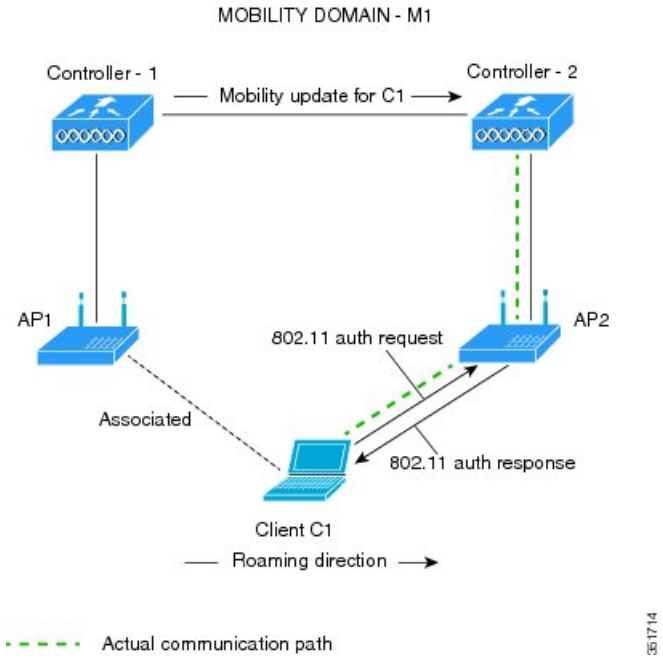
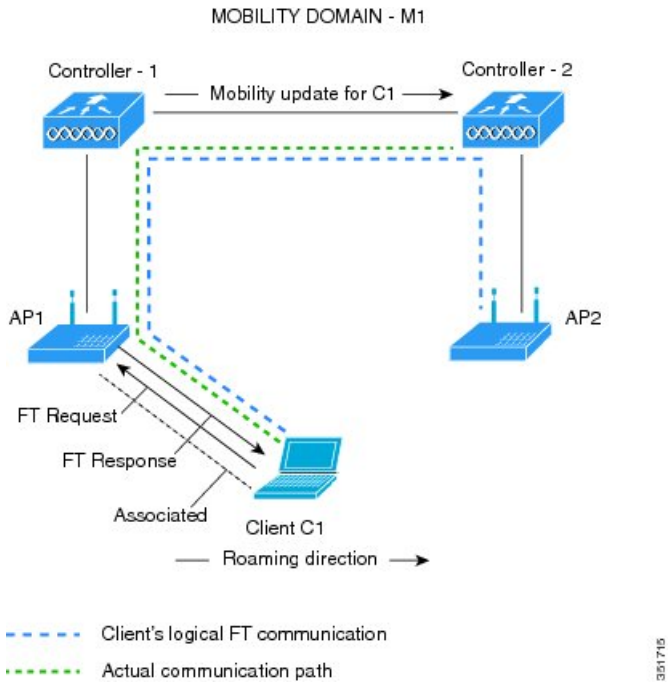


Figure 11: Message exchanges when over–the–DS client roaming is configured



Restrictions for 802.11r Fast Transition

Consider these restrictions when implementing 802.11r Fast Transition:

- EAP LEAP method is not supported.
- Traffic Specification (TSPEC) is not supported for 802.11r fast roaming. Therefore, RIC IE handling is not supported.
- If WAN link latency exists, fast roaming is also delayed. Voice or data maximum latency should be verified. The Cisco WLC handles 802.11r Fast Transition authentication requests during roaming for both Over-the-Air and Over-the-DS methods.
- Legacy clients cannot associate with a WLAN that has 802.11r enabled if the driver of the supplicant that is responsible for parsing the Robust Security Network Information Exchange (RSN IE) is old and not aware of the additional AKM suites in the IE. Due to this limitation, clients cannot send association requests to WLANs. These clients, however, can still associate with non-802.11r WLANs. Clients that are 802.11r-capable can associate as 802.11i clients on WLANs that have both 802.11i and 802.11r Authentication Key Management Suites enabled.

  The workaround is to enable or upgrade the driver of the legacy clients to work with the new 802.11r AKMs, after which the legacy clients can successfully associate with 802.11r-enabled WLANs.

  Another workaround is to have two SSIDs with the same name, but with different security settings (FT and non-FT).
- Fast Transition resource–request protocol is not supported because clients do not support this protocol. Also, the resource–request protocol is an optional protocol.
- To avoid any Denial of Service (DoS) attack, each Cisco WLC allows a maximum of three Fast Transition handshakes with different APs.
- Non-802.11r–capable devices will not be able to associate with FT-enabled WLAN.
- We recommend 802.11r FT Over-the-Air roaming for FlexConnect deployments.

Monitor 802.11r fast transition

These commands can be used to monitor 802.11r Fast Transition:

Command	Description
show WLAN name *WLAN-name*	Displays a summary of the configured parameters on the WLAN.

Command	Description
show wireless client mac-address *mac-address*	Displays the summary of the 802.11r authentication key management configuration on a client. . . . . . . Client Capabilities CF Pollable : Not implemented CF Poll Request : Not implemented Short Preamble : Not implemented PBCC : Not implemented Channel Agility : Not implemented Listen Interval : 15 Fast BSS Transition : Implemented Fast BSS Transition Details : Client Statistics: Number of Bytes Received : 9019 Number of Bytes Sent : 3765 Number of Packets Received : 130 Number of Packets Sent : 36 Number of EAP Id Request Msg Timeouts : 0 Number of EAP Request Msg Timeouts : 0 Number of EAP Key Msg Timeouts : 0 Number of Data Retries : 1 Number of RTS Retries : 0 Number of Duplicate Received Packets : 1 Number of Decrypt Failed Packets : 0 Number of Mic Failured Packets : 0 Number of Mic Missing Packets : 0 Number of Policy Errors : 0 Radio Signal Strength Indicator : -48 dBm Signal to Noise Ratio : 40 dB . . . . . .

Configure 802.11r Fast Transition in an open WLAN (CLI)

Enable 802.11r Fast Transition capabilities on an open WLAN to improve roaming performance for wireless clients.

802.11r Fast Transition allows wireless clients to roam between access points more quickly by pre-authenticating with target access points. This configuration is performed on an open WLAN without WPA security.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device# wlan profile-name
```

Example:

```
Device# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Associate the client VLAN to the WLAN.

Example:

```
Device(config-wlan)# client vlan vlan-id
```

Example:

```
Device(config-wlan)# client vlan 0120
```

Step 4 Disable WPA security.

Example:

```
Device(config-wlan)# no security wpa
```

Step 5 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Step 6 Disable WPA2 security.

Example:

```
Device(config-wlan)# no security wpa wpa2
```

Step 7 Disable WPA2 ciphers for AES.

Example:

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

Step 8 Enable 802.11r Fast Transition parameters.

Example:

```
Device(config-wlan)# security ft
```

Step 9 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 10 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Alternatively, you can also press Ctrl-z to exit global configuration mode

---

The WLAN is now configured with 802.11r Fast Transition enabled, allowing wireless clients to perform faster roaming between access points.

Configure 802.11r BSS Fast Transition on a Dot1x security enabled WLAN (CLI)

Enable 802.11r BSS Fast Transition on a WLAN with 802.1x authentication to provide seamless roaming for wireless clients.

802.11r Fast Transition reduces the time required for wireless clients to roam between access points by enabling pre-authentication and key caching. This configuration is particularly useful in environments where clients need to maintain connectivity while moving between coverage areas.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration submode.

Example:

```
Device# wlan profile-name
```

```
Device# wlan wlan-bss-ft
```

The *profile-name* is the profile name of the configured WLAN.

Step 3 Associate the client VLAN to this WLAN.

Example:

```
Device(config-wlan)# client vlan vlan-name
```

Example:

```
Device(config-wlan)# client vlan 0120
```

Step 4 Enable security authentication list for dot1x security.

Example:

```
Device(config-wlan)# security dot1x authentication-list default
```

The configuration is similar for all dot1x security WLANs.

Step 5 Enable 802.11r Fast Transition on the WLAN.

Example:

```
Device(config-wlan)# security ft
```

Step 6 Enable 802.1x security on the WLAN.

Example:

```
Device(config-wlan)# security wpa akm ft dot1x
```

Step 7 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 8 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Alternatively, you can also press Ctrl-z to exit global configuration mode

Configure 802.11r Fast Transition on a PSK security–enabled WLAN (CLI)

Enable 802.11r Fast Transition capability on a WLAN that uses PSK security to improve roaming performance for wireless clients.

802.11r Fast Transition reduces the time required for client devices to roam between access points by pre-authenticating with neighboring APs. This configuration is performed on WLANs that use pre-shared key (PSK) authentication.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter WLAN configuration mode.

Example:

```
Device(config)# wlan wlan-profile-name
```

Step 3 Associate the client VLAN to this WLAN.

Example:

```
Device(config-wlan)# client vlan vlan-name
```

Example:

```
Device(config-wlan)# client vlan 0120
```

Step 4 Disable security AKM for dot1x.

Example:

```
Device(config-wlan)# no security wpa akm dot1x
```

Step 5 Configure Fast Transition PSK support.

Example:

```
Device(config-wlan)# security wpa akm ft psk
```

Step 6 Configure PSK AKM shared key.

Example:

```
Device(config-wlan)# security wpa akm psk set-key {ascii {0 | 8} | hex {0 | 8}} key-value
```

Example:

```
Device(config-wlan)# security wpa akm psk set-key ascii 0 test
```

Step 7 Configure 802.11r Fast Transition.

Example:

```
Device(config-wlan)# security ft
```

Step 8 Enable the WLAN.

Example:

```
Device(config-wlan)# no shutdown
```

Step 9 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The WLAN is now configured with 802.11r Fast Transition support using PSK authentication, enabling faster client roaming between access points.

Disable 802.11r fast transition (GUI)

Disable 802.11r Fast Transition functionality on a WLAN configuration to prevent automatic roaming assistance between access points.

Use this procedure when you need to turn off Fast Transition capabilities for a specific WLAN. Note that fast transition cannot be enabled or disabled if you have configured an SSID with Open Authentication.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **WLANs** page, click the WLAN name.

Step 3 In the **Edit WLAN** window, click the **Security** > **Layer2** tab.

Step 4 From the **Fast Transition** drop-down list, choose **Disabled**.

Note
You cannot enable or disable Fast Transition, if you have configured an SSID with Open Authentication.

Step 5 Click **Update & Apply to Device**.

The 802.11r Fast Transition feature is disabled for the selected WLAN, and the configuration changes are applied to the device.

Disable 802.11r Fast Transition (CLI)

Disable 802.11r Fast Transition on WLANs to prevent Fast Transitions in 802.11r-enabled networks.

Use this procedure when you need to prevent Fast Transitions in 802.11r-enabled WLANs using the command line interface.

Procedure

Step 1 Enter the global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Enter the WLAN configuration submode.

Example:

```
Device(config)# wlan profile-name
```

```
Device(config)# wlan wlan-test4
```

Step 3 Disable the FT AKMs that are enabled before disabling FT.

Example:

```
Device(config-wlan)# no security wpa akm ft dot1x
```

The available FT AKMs are: SAE-FT, SAE-EXT-FT and 802.1x-FT.

Step 4 Disable 802.11r Fast Transition on the WLAN.

Example:

```
Device(config-wlan)# no security ft {adaptive | over-the-ds | reassociation-timeout
timeout-in-seconds}
```

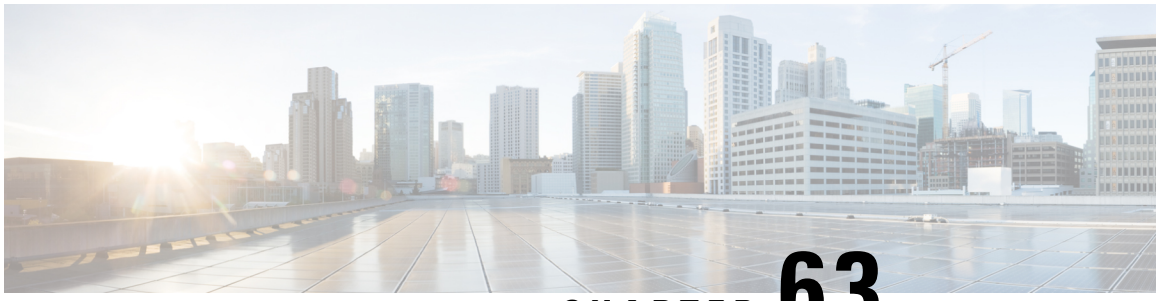
```
Device(config-wlan)# no security ft over-the-ds
```

Step 5 Return to privileged EXEC mode.

Example:

```
Device(config)# end
```

802.11r Fast Transition is disabled on the WLAN and Fast Transitions will no longer occur for clients connecting to this network.



CHAPTER 63

Assisted Roaming

- 802.11k neighbor list and assisted roaming, on page 663
- Restrictions for assisted roaming, on page 664
- How to Configure Assisted Roaming, on page 664
- Verify assisted roaming, on page 666
- Configuration examples for assisted roaming, on page 666

802.11k neighbor list and assisted roaming

The 802.11k neighbor list and assisted roaming is a wireless roaming optimization feature that

- allows an AP to inform 802.11k-capable clients of neighboring BSSIDs (APs in the same SSID)
- helps clients optimize their scanning and roaming behavior, and
- includes Assisted Roaming Prediction Optimization for non-802.11k clients to discourage roaming to suboptimal APs.

Prediction based roaming for non-802.11k clients

You can optimize roaming for non-802.11k clients by generating a prediction neighbor list for each client without sending an 802.11k neighbor list request. When prediction based roaming enables a WLAN, after each successful client association/re-association, the same neighbor list optimization applies on the non-802.11k client to generate and store the neighbor list in the mobile station software data structure. Clients at different locations have different lists because the client probes are seen with different RSSI values by the different neighbors as the clients usually probe before any association or re-association. This list is created with the most updated probe data and predicts the next AP that the client is likely to roam to.

The wireless infrastructure discourages clients from roaming to those less desirable neighbors by denying association if the association request to an AP does not match the entries on the stored prediction neighbor list.

Key configuration parameters for prediction based roaming:

- Denial count: Maximum number of times a client is refused association.
- Prediction threshold: Minimum number of entries required in the prediction list for the assisted roaming feature to activate.

Note We recommend not configuring two SSIDs with the same name in the controller, which may cause roaming issues.

For more information, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/8-5/Enterprise-Mobility-8-5-Design-Guide/Enterprise_Mobility_8-5_Deployment_Guide/Chapter-11.html#pgfId-1140097.

Restrictions for assisted roaming

- This feature is supported only on 802.11n capable indoor access points. For a single band configuration, a maximum of 6 neighbors are visible in a neighbor list. For dual band configuration, a maximum of 12 neighbors are visible.
- You can configure assisted roaming only using the CLI.

How to Configure Assisted Roaming

Configure assisted roaming (CLI)

Enable assisted roaming to improve client roaming performance by providing neighbor AP information and prediction capabilities.

Assisted roaming helps wireless clients make better roaming decisions by providing neighbor lists, dual-band lists, and prediction features. This configuration improves overall wireless network performance and client experience.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure neighbor floor label bias.

Example:

```
Device(config)# wireless assisted-roaming floor-bias dBm-value
```

Example:

```
Device(config)# wireless assisted-roaming floor-bias 20
```

The valid range is from 5 to 25 dBm, and the default value is 15 dBm.

Step 3 Enter WLAN configuration submode.

Example:

```
Device(config)# wlan wlan-id
```

Example:

```
Device(config)# wlan wlan1
```

The *wlan-id* is the profile name of the configured WLAN.

Step 4 Configure an 802.11k neighbor list for a WLAN.

Example:

```
Device(wlan)# assisted-roaming neighbor-list
```

By default, assisted roaming is enabled on the neighbor list when you create a WLAN. The **no** form of the command disables assisted roaming neighbor list.

Step 5 Configure a dual-band 802.11k dual list for a WLAN.

Example:

```
Device(wlan)# assisted-roaming dual-list
```

By default, assisted roaming is enabled on the dual list when you create a WLAN. The **no** form of the command disables assisted roaming dual list.

Step 6 Configure assisted roaming prediction list feature for a WLAN.

Example:

```
Device(wlan)# assisted-roaming prediction
```

By default, the assisted roaming prediction list is disabled.

Note
A warning message is displayed and load balancing is disabled for the WLAN if load balancing is already enabled for the WLAN.

Step 7 Configure the minimum number of predicted APs required for the prediction list feature to be activated.

Example:

```
Device# wireless assisted-roaming prediction-minimum count
```

The default value is 3.

Note
If the number of the AP in the prediction assigned to the client is less than the number that you specify, the assisted roaming feature will not apply on this roam.

Step 8 Configure the maximum number of times a client can be denied association if the association request is sent to an AP that does not match any AP on the prediction.

Example:

```
Device# wireless assisted-roaming denial-maximum count
```

Example:

```
Device# wireless assisted-roaming denial-maximum 8
```

The valid range is from 1 to 10, and the default value is 5.

Step 9 Exit configuration mode and return to privileged EXEC mode.

Example:

```
Device(config)# end
```

Assisted roaming is now configured with the specified parameters, enabling improved client roaming performance through neighbor lists, dual-band support, and prediction capabilities.

Verify assisted roaming

This command can be used to verify assisted roaming configured on a WLAN:

Command	Description
show WLAN id *WLAN-id*	Displays the WLAN parameters on the WLAN.

Configuration examples for assisted roaming

Configuration examples for assisted roaming demonstrate the necessary commands and syntax to configure neighbor floor label bias, disable neighbor list on specific WLANs, configure prediction lists, and set prediction thresholds and maximum denial counts.

This example shows how to configure the neighbor floor label bias:

```
Device# configure terminal
Device(config)# wireless assisted-roaming floor-bias 10
Device(config)# end
Device# show wlan id 23
```

This example shows how to disable neighbor list on a specific WLAN:

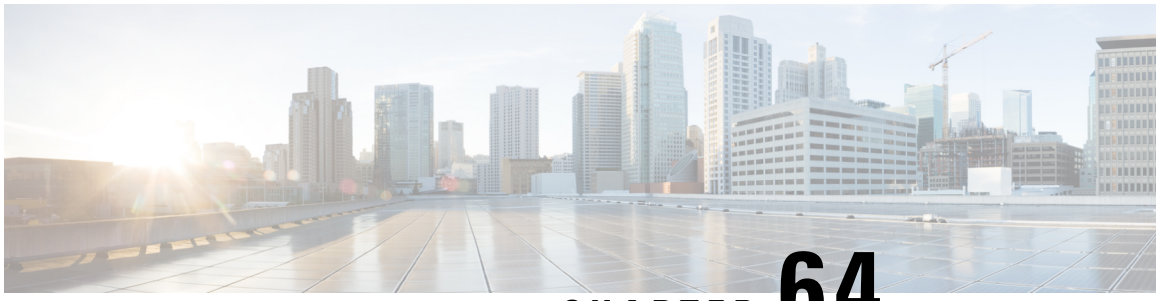
```
Device# configure terminal
Device(config)# wlan test1
Device(config (wlan)# no assisted-roaming neighbor-list
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list on a specific WLAN:

```
Device# configure terminal
Device(config)# wlan test1
Device(config)(wlan)# assisted-roaming prediction
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list based on assisted roaming prediction threshold and maximum denial count on a specific WLAN:

```
Device# configure terminal
Device(config)# wireless assisted-roaming prediction-minimum 4
Device(config)# wireless assisted-roaming denial-maximum 4
Device(config)# wlan wlan-name
Device(config)(wlan)# end
Device# show wlan id 23
```



CHAPTER 64

802.11v

- 802.11v amendment, on page 667

802.11v amendment

The 802.11v amendment is a wireless network standard that

- describes numerous enhancements to wireless network management
- enables Network assisted Power Savings which helps clients to improve the battery life by enabling them to sleep longer, and
- enables Network assisted Roaming which enables the WLAN to send requests to associated clients, advising the clients as to better APs to associate to, for both load balancing and in directing poorly connected clients.

802.11v network assisted power savings

Wireless devices consume battery to maintain their connection to the clients, in several ways:

- By waking up at regular intervals to listen to the access point beacons containing a DTIM, which indicates buffered broadcast or multicast traffic that the access point delivers to the clients.
- By sending null frames to the access points, in the form of keepalive messages– to maintain connection with access points.
- Devices also periodically listen to beacons (even in the absence of DTIM fields) to synchronize their clock to that of the corresponding access point.

All these processes consume battery and this consumption particularly impacts devices (such as Apple), because these devices use a conservative session timeout estimation, and therefore, wake up often to send keepalive messages. The 802.11 standard, without 802.11v, does not include any mechanism for the controller or the access points to communicate to wireless clients about the session timeout for the local client.

To save the power of clients due to the mentioned tasks in wireless network, these features in the 802.11v standard are used:

- Directed multicast service
- Base station subsystem (BSS) max idle period

Directed multicast service:

Using Directed Multicast Service (DMS), the client requests the access point to transmit the required multicast packet as unicast frames. This allows the client to receive the multicast packets it has ignored while in sleep mode and also ensures Layer 2 reliability. Furthermore, the unicast frame is transmitted to the client at a potentially higher wireless link rate which enables the client to receive the packet quickly by enabling the radio for a shorter duration, thus also saving battery power. Since the wireless client also does not have to wake up at each DTIM interval in order to receive multicast traffic, longer sleeping intervals are allowed.

BSS max idle period:

The BSS max idle period is the timeframe during which an access point (AP) does not disassociate a client due to nonreceipt of frames from the connected client. This helps ensure that the client device does not send keepalive messages frequently. The idle period timer value is transmitted using the association and reassociation response frame from the access point to the client. The idle time value indicates the maximum time that a client can remain idle without transmitting any frame to an access point. As a result, the clients remain in sleep mode for a longer duration without transmitting the keepalive messages often. This in turn contributes to saving battery power.

Prerequisites for configuring 802.11v

Requirements for configuring 802.11v support in your wireless network environment.

- Applies for Apple clients like Apple iPad, iPhone, and so on, that run on Apple iOS version 7 or later.
- Supports local mode; also supports FlexConnect access points in central authentication modes only.

Restrictions for 802.11v

Client needs to support 802.11v BSS Transition.

802.11v BSS transition management

802.11v BSS transition management

- enables clients to request better AP options before roaming
- allows APs to suggest client transitions for load balancing, and
- supports optimized roaming based on RSSI and rate requirements.

802.11v BSS transition scenarios

802.11v BSS Transition is applied in these three scenarios:

- Solicited request: Client can send an 802.11v Basic Service Set (BSS) Transition Management Query before roaming for a better option of AP to reassociate with.
- Unsolicited Load Balancing request: If an AP is heavily loaded, it sends out an 802.11v BSS Transition Management Request to an associated client.
- Unsolicited Optimized Roaming request: If a client's RSSI and rate do not meet the requirements, the corresponding AP sends out an 802.11v BSS Transition Management Request to this client.

Note 802.11v BSS Transition Management Request is a suggestion (or advice) given to a client, which the client can choose to follow or ignore. To force the task of disassociating a client, turn on the disassociation-imminent function. This disassociates the client after a period if the client is not reassociated to another AP.

Configure 802.11v BSS transition management (GUI)

Enable and customize 802.11v BSS transition management features for a particular WLAN to improve client roaming, battery life, and network efficiency.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add** to create WLANs.

The **Add WLAN** page is displayed.

Step 3 In the **Advanced** tab and **11v BSS Transition Support** section, select the **BSS Transition** check box to enable BSS transition per WLAN.

Step 4 Enable the **Dual Neighbor List** check box to include the neighbors of other radio slots of the same AP in the BSS transition response.

Note
This is applicable only for 2.4 GHz and 5 GHz radio slots.

Step 5 Enable the **BSS Max Idle Service** check box to help clients and APs efficiently decide how long to remain associated when no traffic is being transmitted. The device uses this information to preserve device battery life.

Step 6 Enable the **BSS Max Idle Protected** check box to enable the AP to accept only authenticated frames (encrypted with Robust Security Network (RSN) information) from the client to reset the BSS Max Idle period counter. Without protected mode, any data or management frame (encrypted or unencrypted) sent by the client will reset the idle timer for the client.

Step 7 Enable the **Directed Multicast Service** check box to request the AP to send a multicast stream as unicast, to any DMS capable client on this WLAN.

Step 8 Click **Save & Apply to Device**.

The selected WLAN is now configured with 802.11v BSS transition management features. Clients benefit from improved roaming and battery conservation, and multicast streams can be delivered more efficiently.

Configure 802.11v BSS transition management (CLI)

Enable 802.11v BSS transition management on a WLAN to improve client roaming and load balancing.

802.11v BSS transition management allows access points to provide roaming assistance to clients by suggesting better access points and triggering client roaming decisions. This feature is applied in scenarios where network optimization and seamless client mobility are required.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure WLAN profile and enter the WLAN profile configuration mode.

Example:

```
Device(config)# wlan profile-name
```

Example:

```
Device(config)# wlan test-wlan
```

Step 3 Shutdown the WLAN profile.

Example:

```
Device(config-wlan)# shutdown
```

Step 4 Configure BSS transition per WLAN.

Example:

```
Device(config-wlan)# bss-transition
```

Step 5 Configure BSS transition disassociation imminent per WLAN.

Example:

```
Device(config-wlan)# bss-transition disassociation-imminent
```

Step 6 Enable the WLAN profile.

Example:

```
Device(config-wlan)# no shutdown
```

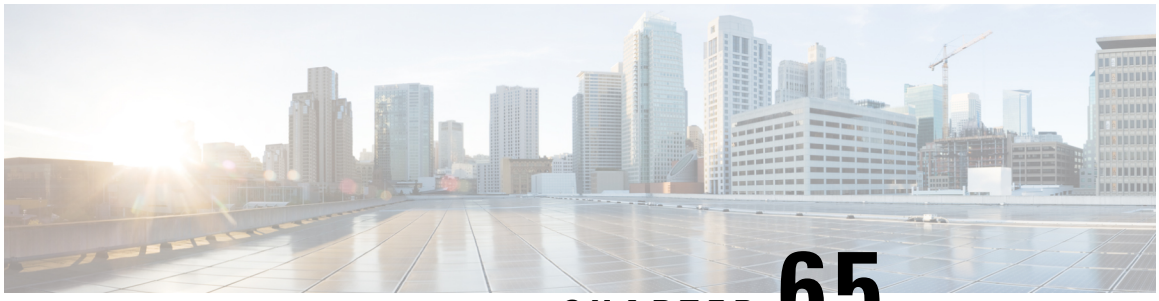
Step 7 Return to privileged EXEC mode.

Example:

```
Device(config-wlan)# end
```

Alternatively, you can press CTRL + Z to exit global configuration mode.

802.11v BSS transition management is now configured on the WLAN, enabling improved client roaming assistance and network optimization.



CHAPTER 65

802.11w

- Protected management frames with 802.11w, on page 671
- Configure 802.11w, on page 675

Protected management frames with 802.11w

A protected management frame is a wireless security feature that

- uses the 802.11w protocol to safeguard management frames
- prevents spoofing and forgery of authentication, de-authentication, association, and disassociation frames, and
- enhances the overall security of Wi-Fi networks by protecting key network management actions from attack.

While data frames can be encrypted, management frames were traditionally sent in the clear, making them vulnerable to interception and forgery. The 802.11w standard addresses this vulnerability by requiring cryptographic protection for certain management frames between client and access point.

An attacker attempting to disconnect a legitimate client by sending forged disassociation frames will fail on networks supporting protected management frames, as only cryptographically validated frames are accepted.

Types of management frames protected by 802.11w

The 802.11w protocol protects certain management frames by using the Protected Management Frames (PMF) service. These frames are classified as robust management frames and include:

- Disassociation frames
- De-authentication frames
- Robust Action frames

Robust Action frames protected by 802.11w include:

- Spectrum Management
- Quality of Service (QoS)
- Direct Link Setup (DLS)

- Block acknowledgement
- Radio Measurement
- Fast Basic Service Set (BSS) Transition
- Security Association (SA) Query
- Protected Dual of Public Action
- Vendor-specific Protected

Frames not included in this list are not protected by 802.11w.

Protections offered by 802.11w

When 802.11w is implemented, these protections are provided:

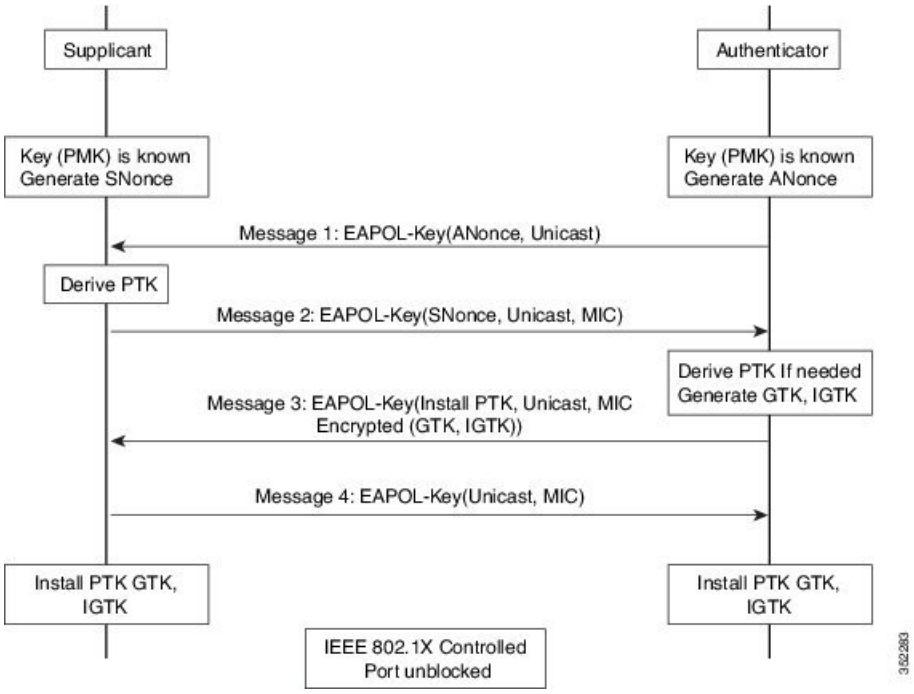
- **Client protection:** The AP adds cryptographic protection to de-authentication and dissociation frames, preventing spoofing in DOS attacks.
- **Infrastructure protection**: Security Association (SA) teardown protection is implemented using Association Comeback Time and SA-Query procedures to prevent spoofed association requests from disconnecting connected clients.

Integrity Group Temporal Key (IGTK)

An integrity group temporal key (IGTK) is a wireless security mechanism that

- protects broadcast and multicast robust management frames.
- uses random values assigned by the authenticator station (such as a wireless LAN controller), and
- secures MAC management protocol data units (MMPDUs) in 802.11w networks.

Figure 12: IGTK exchange in four-way handshake



802.11w introduced IGTKs to enhance the security of management frames in wireless networks.

How IGTK is used in 802.11w management frame protection

When you enable management frame protection, the AP encrypts the group temporal key (GTK) and IGTK values in an Extensible Authentication Protocol over LAN-Key (EAPoL-Key) frame. The AP includes this frame in the third message of the four-way handshake.

- IGTK is exchanged during the four-way handshake process.
- If the AP changes the GTK later, it sends the new GTK and the new IGTK to your client device using the Group Key Handshake

Imagine a wireless network as a secured meeting room, with announcements broadcast to everyone inside. The IGTK acts like a special group password that lets only authorized members hear these important messages. The authenticator (such as a wireless LAN controller) is like the meeting organizer, who gives each member a random, unique password at the door. When the password changes for added security, the organizer discreetly shares the new password with all members so that only legitimate participants can continue to hear future group announcements.

Broadcast or multicast integrity protocol (BIP)

A broadcast or multicast integrity protocol (BIP) is a wireless security mechanism that

- ensures data integrity of broadcast and multicast robust management frames
- provides replay protection for these frames after you establish an Integrity Group Temporal Key Security Association (IGTKSA), and

- adds a message integrity code (MIC) calculated using the shared IGTK key.

SA teardown protection

SA teardown protection is a wireless network security mechanism that

- prevents spoofed or replay attacks from disconnecting already associated clients
- uses Association Comeback Time and an SA-Query procedure to verify the authenticity of association requests, and
- ensures the AP only accepts new associations after the original security association is proven invalid.

How association comeback time and SA query procedures work

This process describes how Association Comeback Time and SA Query procedures protect wireless client sessions from replay-based association teardown attacks."

Summary

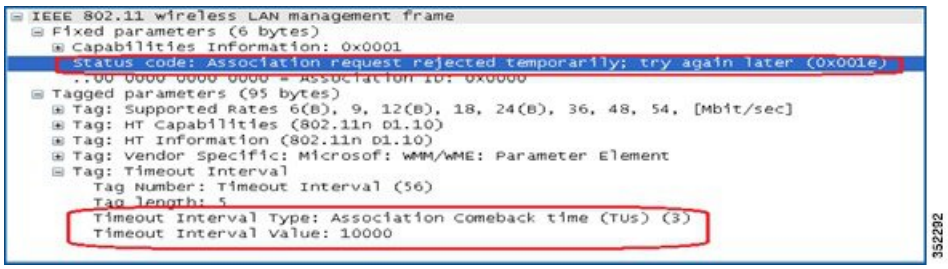
The key components involved in the process are:

- AP: implements security association (SA) teardown protection and manages association requests."
- Client device: Maintains a security association and sends and receives association and SA Query frames.
- SA Teardown Protection Mechanism: Provides the logic for handling replay and spoofed association attempts.

In this process, the AP and client device exchange association and SA Query frames. This exchange validates security associations and prevents unauthorized session teardown.

Workflow

Figure 13: Association Reject with Comeback Time



These stages describe how the Association Comeback Time and SA Query procedures operate to protect client sessions:

1. When an AP receives an Association Request from a client with an existing valid security association (SA) negotiated with 802.11w, the AP rejects the request with status code 30 ("Association request rejected temporarily; try again later") and sends an Association Comeback Time.
2. The AP does not modify the existing association during the comeback interval.

3. If no ongoing SA Query session with the client exists, the AP sends SA Query requests repeatedly until it receives a valid SA Query response or the comeback time expires.
4. Receiving a matching SA Query response or a valid protected frame indicates a valid SA. The AP may then allow a new association attempt without more SA Query cycles.

Result

This process ensures that spoofed requests cannot disconnect valid clients, protecting against replay-based association teardown attacks.

Prerequisites for 802.11w

- To configure 802.11w feature for optional and mandatory, you must have WPA and AKM configured.



Note The RNS (Robust Secure Network) IE must be enabled with an AES Cipher.

Restrictions for 802.11w

- You cannot use 802.11w with open WLANs, Wired Equivalent Privacy (WEP)-encrypted WLANs, or Temporal Key Integrity Protocol (TKIP)-encrypted WLANs.
- You can use 802.11w with Protected Management Frames (PMF) for non-Apple clients. For Apple iOS version 11 and earlier, request a fix from Apple to resolve association issues
- When clients do not use 802.11w PMF, the controller ignores disassociation frames or deauthentication frames they send. If a client uses PMF, its entry is deleted immediately when the controller receives such a frame. This process helps prevent denial-of-service attacks by malicious devices, since frames without PMF are not secure.

Configure 802.11w

Configure 802.11w (GUI)

To protect management traffic against spoofing and replay attacks, configure 802.11w settings after enabling WPA and AKM.

Enable protected management frames (PMF) on a WLAN to strengthen security for management communications.

Before you begin

Ensure WPA and AKM are configured on the target WLAN.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add** to create WLANs.

The **Add WLAN** page is displayed.

Step 3 In the **Security** > **Layer2** tab, navigate to the **Protected Management Frame** section.

Step 4 Choose **PMF** as *Disabled*, *Optional*, or *Required*. By default, the PMF is *disabled*.

If you choose **PMF** as *Optional* or *Required*, you can view these fields:

- **Association Comeback Timer**—Enter a value between 1 and 10 seconds to configure 802.11w association comeback time.
- **SA Query Time**—Enter a value between 100 to 500 milliseconds. This is required for clients to negotiate 802.11w PMF protection on a WLAN.

Step 5 Click **Save & Apply to Device**.

802.11w PMF is enabled with the parameters you selected. This strengthens the security of management frames on the WLAN.

Configure 802.11w (CLI)

Configuring 802.11w improves security for management frames on wireless LANs.

Enable 802.11w Protected Management Frames (PMF) on a WLAN using CLI commands.

Before you begin

- Configure WPA.
- Configure AKM (Authentication and Key Management) on the WLAN.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a WLAN and enters configuration mode.

Example:

```
Device(config)# wlan profile-name wlan-id ssid
```

Step 3 Configure 802.1x support using the **security wpa akm dot1x-sha256** command.

Example:

```
Device(config-wlan)#security wpa akm dot1x-sha256
```

Step 4 Configure the 802.11w association comeback time.

Example:

```
Device(config-wlan)# security pmf association-comeback comeback-interval
```

Example:

```
Device(config-wlan)# security pmf association-comeback 10
```

Step 5 Require clients to negotiate 802.11w PMF protection on a WLAN.

Example:

```
Device(config-wlan)# security pmf mandatory
```

Step 6 Configure time interval identified in milliseconds before which the SA query response is expected.

Example:

```
Device(config-wlan)# security pmf saquery-retry-time timeout
```

Example:

```
Device(config-wlan)# security pmf saquery-retry-time 100
```

If the device does not get a response, another SQ query is tried.

802.11w Protected Management Frames are enabled and mandatory on the specified WLAN, providing improved protection of management frames.

Disable 802.11w

Disabling 802.11w may be necessary for compatibility with legacy or non-compliant devices.

Prevent wireless clients from using 802.11w Protected Management Frames on a specified WLAN.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Configure a WLAN and enters configuration mode.

Example:

```
Device(config)# wlan profile-name wlan-id ssid
```

Step 3 Disable 802.1x support using the **no security wpa akm dot1x-sha256** command.

Example:

```
Device(config-wlan)# no security wpa akm dot1x-sha256
```

Step 4 Configure the 802.11w association comeback time.

Example:

```
Device(config-wlan)# security pmf association-comeback comeback-interval
Device(config-wlan)# security pmf association-comeback 10
```

Step 5 Disable client negotiation of 802.11w PMF protection on a WLAN.

Example:

```
Device(config-wlan)# no security pmf mandatory
```

Step 6 Disable SQ query retry.

Example:

```
Device(config-wlan)# no security pmf saquery-retry-time 100
```

802.11w PMF protection is disabled for the specified WLAN.

Monitor 802.11w

Use these commands to monitor 802.11w.

Procedure

Step 1 Display the WLAN parameters on the WLAN.

The PMF parameters are displayed.

```
Device# show wlan name wlan-name
                        . . . .
                        . . . .
                        Auth Key Management
                        802.1x                                  : Disabled
                        PSK                                     : Disabled
                        CCKM                                    : Disabled
                        FT dot1x                                : Disabled
                        FT PSK                                  : Disabled
                        FT SAE                                  : Disabled
                        Dot1x-SHA256                            : Enabled
                        PSK-SHA256                              : Disabled
                        SAE                                     : Disabled
                        OWE                                     : Disabled
                        SUITEB-1X                               : Disabled
                        SUITEB192-1X                            : Disabled
                        CCKM TSF Tolerance                              : 1000
                        FT Support                                      : Adaptive
                        FT Reassociation Timeout                : 20
                        FT Over-The-DS mode                     : Enabled
                        PMF Support                                     : Required
                        PMF Association Comeback Timeout        : 1
                        PMF SA Query Time                       : 500
                        . . . .
                        . . . .
```

Step 2 Display the summary of the 802.11w authentication key management configuration on a client.

```
Device# show wireless client mac-address mac-address detail
                        . . . .
                        . . . .
                        Policy Manager State: Run
                        NPU Fast Fast Notified : No
                        Last Policy Manager State : IP Learn Complete
                        Client Entry Create Time : 497 seconds
                        Policy Type : WPA2
                        Encryption Cipher : CCMP (AES)
                        Authentication Key Management : 802.1x-SHA256
                        Encrypted Traffic Analytics : No
                        Management Frame Protection : No
                        Protected Management Frame - 802.11w : Yes
                        EAP Type : LEAP
                        VLAN : 39
                        Multicast VLAN : 0
                        Access VLAN : 39
                        Anchor VLAN : 0
                        WFD capable : No
                        Manged WFD capable : No
                        . . . .
                        . . . .
```