



Cisco Aironet Wave 2 and Catalyst Wi-Fi6/6E Access Point Command Reference, Release 8.9

First Published: 2019-06-07

Last Modified: 2021-02-16

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 NON-INFRINGEMENT 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.

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

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)

© 2019–2022 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	ix
Audience	ix
Document Conventions	ix
Related Documentation	xii
Obtaining Documentation and Submitting a Service Request	xii

CHAPTER 1

Using the Command Line Interface	1
Understanding Command Modes	2
Understanding Abbreviated Commands	3
Understanding no Forms of Commands	4
Understanding CLI Error Messages	5
Configuring the Terminal	6
Recalling Commands	7
Accessing the CLI	8

CHAPTER 2

capwap Commands	9
capwap ap	10
capwap ap auth-token	11
capwap ap erase	12
capwap ap ethernet	13
capwap ap hostname	14
capwap ap ip	15
capwap ap lag	16
capwap ap mesh strict-wired-uplink	17
capwap ap mode	18
capwap ap restart	19

CHAPTER 3**clear Commands 21**

- clear avc nbar 22
- clear counters 23
- clear cts 24
- clear datapath 25
- clear dot11 26
- clear logging 27

CHAPTER 4**config Commands 29**

- config ap address 30
- config ap client-trace 31
- config ap client-trace filter 33
- config ap client-trace output 34
- config boot baudrate 35
- config boot break 36
- config boot crashkernel 37
- config boot debug-memory 38
- config boot manual 39
- config boot path 40
- config cts debug enforcement host_ip 41
- config cts debug enforcement rate 42
- config cts debug enforcement permissions 43
- config cts debug enforcement protocol 44

CHAPTER 5**debug Commands 45**

- debug arp 47
- debug ble 48
- debug capwap client 49
- debug capwap client avc 50
- debug cdp 51
- debug cleanair 52
- debug dhcp 53
- debug dot11 54

debug dot11 client datapath	55
debug dot11 client level	56
debug dot11 driver slot	57
debug dot11 firmware	59
debug dot11 sensor	60
debug dtls client	61
debug ethernet	62
debug flexconnect	63
debug lldp	64
debug memory	65
debug memory pool	66
debug memory pool alloc	67
debug memory pool free	68
debug mesh	69
debug mesh adjacency	70
debug mesh path-control	71
debug rrm neighbor	72
debug rrm reports	73
debug sip	74
debug wips	75
debug authentication interface	76
debug process memory	77
debug traffic	78
debug tunnel	79
debug client trace	80
no	81
traceroute	82
undebug	83

CHAPTER 6

show Commands	85
show ap client-trace status	87
show arp	88
show avc cft	89
show avc nbar	90

show avc netflow flows	91
show avc status	92
show boot	93
show capwap	94
show capwap client	95
show capwap client trace	96
show capwap ids sig	97
show cdp	98
show class-map	99
show cleanair debug	100
show client statistics	101
show clock	102
show configuration	103
show controller ble	104
show controllers dot11Radio	105
show controllers nss status	106
show controllers wired	107
show crypto	108
show debug	109
show dhcp	110
show dot11 qos	111
show filesystems	112
show flash	113
show flexconnect	114
show flexconnect ocap firewall	115
show flexconnect wlan	116
show interfaces dot11Radio	117
show interfaces network	118
show interfaces wired	119
show inventory	120
show ip	121
show lacp	122
show logging	123
show memory	124

show policy-map	126
show processes	127
show processes memory	128
show rrm	130
show rrm rogue containment	131
show rrm rogue detection	132
show running-config	134
show spectrum	136
show tech-support	137
show version	138
show trace dot11_chn	139
show trace	140
show wips	141

CHAPTER 7**System Management Commands 143**

ap-type	144
archive	145
copy	146
delete	147
disable	148
enable	149
exec-timeout	150
logging	151
more	152
reload	153
terminal	154



Preface

This preface describes the audience, organization, and conventions of the Cisco Aironet Wave 2 Access Point Command Reference. It also provides information about how to obtain other documentation.

- [Audience, on page ix](#)
- [Document Conventions, on page ix](#)
- [Related Documentation, on page xii](#)
- [Obtaining Documentation and Submitting a Service Request, on page xii](#)

Audience

This publication is for experienced network administrators who configure and maintain Cisco Aironet Wave 2 Access Points.



Note Usage of **test** commands may cause system disruption such as unexpected reboot of the Cisco AP. Therefore, we recommend that you use the **test** commands on Cisco APs for debugging purposes with the help of Cisco Technical Assistance Center (TAC) personnel.

Document Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in bold font .
<i>italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in <i>italic font</i> .
[]	Elements in square brackets are optional.
{x y z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x y z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.

Convention	Indication
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
<code>courier font</code>	Terminal sessions and information the system displays appear in <code>courier font</code> .
<>	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.



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 perform an action that could result in equipment damage or loss of data.



Warning This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. (To see translations of the warnings that appear in this publication, refer to the appendix "Translated Safety Warnings.")

Warning Title	Description
Waarschuwing	Dit waarschuwingssymbool betekent gevaar. U verkeert in een situatie die lichamelijk letsel kan veroorzaken. Voordat u aan enige apparatuur gaat werken, dient u zich bewust te zijn van de bij elektrische schakelingen betrokken risico's en dient u op de hoogte te zijn van standaard maatregelen om ongelukken te voorkomen. (Voor vertalingen van de waarschuwingen die in deze publicatie verschijnen, kunt u het aanhangsel "Translated Safety Warnings" (Vertalingen van veiligheidsvoorschriften) raadplegen.)
Varoitus	Tämä varoitusmerkki merkitsee vaaraa. Olet tilanteessa, joka voi johtaa ruumiinvammaan. Ennen kuin työskentelet minkään laitteiston parissa, ota selvää sähkökytkentöihin liittyvistä vaaroista ja tavanomaisista onnettomuuksien ehkäisykeinoista. (Tässä julkaisussa esiintyvien varoitusten käännökset löydät liitteestä "Translated Safety Warnings" (käännetyt turvallisuutta koskevat varoitukset).)

Warning Title	Description
Attention	Ce symbole d'avertissement indique un danger. Vous vous trouvez dans une situation pouvant entraîner des blessures. Avant d'accéder à cet équipement, soyez conscient des dangers posés par les circuits électriques et familiarisez-vous avec les procédures courantes de prévention des accidents. Pour obtenir les traductions des mises en garde figurant dans cette publication, veuillez consulter l'annexe intitulée « Translated Safety Warnings » (Traduction des avis de sécurité).
Warnung	Dieses Warnsymbol bedeutet Gefahr. Sie befinden sich in einer Situation, die zu einer Körperverletzung führen könnte. Bevor Sie mit der Arbeit an irgendeinem Gerät beginnen, seien Sie sich der mit elektrischen Stromkreisen verbundenen Gefahren und der Standardpraktiken zur Vermeidung von Unfällen bewusst. (Übersetzungen der in dieser Veröffentlichung enthaltenen Warnhinweise finden Sie im Anhang mit dem Titel "Translated Safety Warnings" (Übersetzung der Warnhinweise).)
Avvertenza	Questo simbolo di avvertenza indica un pericolo. Si è in una situazione che può causare infortuni. Prima di lavorare su qualsiasi apparecchiatura, occorre conoscere i pericoli relativi ai circuiti elettrici ed essere al corrente delle pratiche standard per la prevenzione di incidenti. La traduzione delle avvertenze riportate in questa pubblicazione si trova nell'appendice, "Translated Safety Warnings" (Traduzione delle avvertenze di sicurezza).
Advarsel	Dette varselsymbolet betyr fare. Du befinner deg i en situasjon som kan føre til personskade. Før du utfører arbeid på utstyr, må du være oppmerksom på de faremomentene som elektriske kretser innebærer, samt gjøre deg kjent med vanlig praksis når det gjelder å unngå ulykker. (Hvis du vil se oversettelser av de advarslene som finnes i denne publikasjonen, kan du se i vedlegget "Translated Safety Warnings" [Oversatte sikkerhetsadvarsler].)
Aviso	Este símbolo de aviso indica perigo. Encontra-se numa situação que lhe poderá causar danos físicos. Antes de começar a trabalhar com qualquer equipamento, familiarize-se com os perigos relacionados com circuitos eléctricos, e com quaisquer práticas comuns que possam prevenir possíveis acidentes. (Para ver as traduções dos avisos que constam desta publicação, consulte o apêndice "Translated Safety Warnings" - "Traduções dos Avisos de Segurança").
¡Advertencia!	Este símbolo de aviso significa peligro. Existe riesgo para su integridad física. Antes de manipular cualquier equipo, considerar los riesgos que entraña la corriente eléctrica y familiarizarse con los procedimientos estándar de prevención de accidentes. (Para ver traducciones de las advertencias que aparecen en esta publicación, consultar el apéndice titulado "Translated Safety Warnings.")
Varning	Denna varningssymbol signalerar fara. Du befinner dig i en situation som kan leda till personskada. Innan du utför arbete på någon utrustning måste du vara medveten om farorna med elkretsar och känna till vanligt förfarande för att förebygga skador. (Se förklaringar av de varningar som förekommer i denna publikation i appendix "Translated Safety Warnings" [Översatta säkerhetsvarningar].)

Related Documentation

- Cisco Access Points—<https://www.cisco.com/c/en/us/products/wireless/access-points/index.html>
- Cisco Wireless Controller Software Documentation—<https://www.cisco.com/c/en/us/support/wireless/wireless-lan-controller-software/tsd-products-support-series-home.html>

Obtaining Documentation and Submitting a Service Request

For information about obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as an RSS feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service. Cisco currently supports RSS Version 2.0.



Using the Command Line Interface

This chapter describes the Cisco Aironet Wave 2 AP command-line interface (CLI) and how to use it to configure your AP.

- [Understanding Command Modes, on page 2](#)
- [Understanding Abbreviated Commands, on page 3](#)
- [Understanding no Forms of Commands, on page 4](#)
- [Understanding CLI Error Messages, on page 5](#)
- [Configuring the Terminal, on page 6](#)
- [Recalling Commands, on page 7](#)
- [Accessing the CLI, on page 8](#)

Understanding Command Modes

The Cisco Aironet Wave 2 AP command line interface is divided into the following two different modes:

- **User EXEC mode**—When you start a session on the AP, you begin in the User EXEC mode. Only a limited subset of the commands are available in this mode. Also, the **show** commands that are available in the User EXEC mode are a subset of the **show** commands that are available in the Privileged EXEC mode.

The user EXEC commands are not saved when the AP is rebooted.

- **Privileged EXEC mode**—In this mode, you will have access to all commands. You are required to enter a password to enter the Privileged EXEC mode.

The commands available to you depend on which mode you are currently in. Enter a question mark (?) at the system prompt to obtain a list of commands available for the command mode you are in. For example, here are the list of User EXEC mode commands available:

```
cisco-wave2-ap>?
Exec mode commands
  enable  Turn on privileged commands
  logout  Logout out from CLI
  ping    Send echo messages
  show    Show running system information
```

Table 1: Command Mode Summary

Mode	Access Method	Prompt	Exit Method	About This Mode
User EXEC	Begin a session with your switch.	cisco-wave2-ap>	Enter logout or quit .	Use this mode to <ul style="list-style-type: none"> • Change terminal settings. • Perform basic tests. • Display system information.
Privileged EXEC	While in user EXEC mode, enter the enable command and enter the password when prompted.	cisco-wave2-ap#	Enter disable to exit.	Use this mode to verify commands that you have entered. Use a password to protect access to this mode.

Understanding Abbreviated Commands

You need to enter only enough characters for the AP to recognize the command as unique.

This example shows how to enter the **show configuration** privileged EXEC command in an abbreviated form:

```
cisco-ap# show conf
```

Understanding no Forms of Commands

While you need to use the **debug** command to enable debugs on many features, the prefix **no** disables debugs on those respective features. For example:

Command to enable debug:

```
cisco-ap# debug client ...
```

Command to disable debug:

```
cisco-ap# no debug client ...
```


Understanding CLI Error Messages

This table lists some error messages that you might encounter while using the CLI to configure your AP.

Table 2: Common CLI Error Messages

Error Message	Meaning	How to Get Help
<code>% Ambiguous command: "show con"</code>	You did not enter enough characters for your AP to recognize the command.	Enter the command again followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
<code>% Incomplete command.</code>	You did not enter all the keywords or values required by this command.	Enter the command again followed by a question mark (?) with a space between the command and the question mark. The possible keywords that you can enter with the command appear.
<code>% Invalid input detected at '^' marker.</code>	You entered the command incorrectly. The caret (^) marks the point of the error.	Enter a question mark (?) to display all the commands that are available in this command mode. The possible keywords that you can enter with the command appear.

Configuring the Terminal

Before you begin

Enter the Privileged EXEC mode.

Procedure

- Configure the number of lines on the screen by entering this command:

terminal length *number-of-lines*

Valid range is 0 to 512. If you enter 0, there will be no pausing.

Example:

```
cisco-ap# terminal length 20
```

- Copy debug output to the current terminal line by entering this command:

terminal monitor

- Disable logging to the current terminal line by entering this command:

terminal monitor disable

- Specify the terminal type by entering this command:

terminal type *type-name*

- Configure the number of characters that should be displayed on a screen line by entering this command:

terminal width *number-of-characters*

Valid range is 0 to 132.

Example:

```
cisco-ap# terminal width 30
```

Recalling Commands

To recall commands from the history buffer, perform one of the actions listed in this table. These actions are optional.



Note The arrow keys function only on ANSI-compatible terminals such as VT100s.

Table 3: Recalling Commands

Action	Result
Press the up arrow key	Recalls commands in the history buffer, beginning with the most recent command. Repeat the key sequence to recall successively older commands.
Press the down arrow key	Returns to more recent commands in the history buffer after recalling commands with the up arrow key. Repeat the key sequence to recall successively more recent commands.

Accessing the CLI

You can access the CLI through a console connection, through Telnet, or by using the browser. Commands you enter in one session are not displayed in the other sessions. Therefore, it is possible to lose track of the session from which you entered commands.



capwap Commands

- [capwap ap](#), on page 10
- [capwap ap auth-token](#), on page 11
- [capwap ap erase](#), on page 12
- [capwap ap ethernet](#), on page 13
- [capwap ap hostname](#), on page 14
- [capwap ap ip](#), on page 15
- [capwap ap lag](#), on page 16
- [capwap ap mesh strict-wired-uplink](#), on page 17
- [capwap ap mode](#), on page 18
- [capwap ap restart](#), on page 19

capwap ap

To configure the primary, secondary and tertiary controllers for the AP, use the **capwap ap** command.

```
capwap ap {primary-base | secondary-base | tertiary-base}
controller-name controller-ip-address
```

Syntax Description		
	primary-base	Configure AP's primary controller
	secondary-base	Configure AP's secondary controller
	tertiary-base	Configure AP's tertiary controller
	<i>controller-name</i>	Name of the controller
	<i>controller-ip-address</i>	IP address of the controller.

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure the primary controller for the AP:

```
cisco-ap# capwap ap primary-base wlc-5520 209.165.200.224
```

capwap ap auth-token

To configure authentication token, use the **capwap ap auth-token** command.

capwap ap auth-token *ssc-token*

Syntax Description	<i>ssc-token</i> SSC token; valid range is 8 to 32 characters				
Command Modes	Privileged EXEC (#)				
Command History	<table><thead><tr><th>Release</th><th>Modification</th></tr></thead><tbody><tr><td>8.1.111.0</td><td>This command was introduced.</td></tr></tbody></table>	Release	Modification	8.1.111.0	This command was introduced.
Release	Modification				
8.1.111.0	This command was introduced.				

Examples

The following example shows how to configure authentication token,:

```
cisco-ap# capwap ap auth-token myauthtoken
```

capwap ap erase

To erase CAPWAP configuration, use the **capwap ap erase** command.

capwap ap erase { **all** | **static-ip** }

Syntax Description

all	Erases all CAPWAP configuration
Note	If the AP is in Bridge mode, then the same Bridge mode is retained after the factory reset of the AP; if the AP is in FlexConnect, Local, Sniffer, or any other mode, then the AP mode is set to Local mode after the factory reset of the AP. If you press the Reset button on the AP and perform a true factory reset, then the AP moves to a cookie configured mode.

static-ip Erase static IP or DNS configuration

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to erase all the CAPWAP configuration on the AP:

```
cisco-ap# capwap ap erase all
```


capwap ap ethernet

To configure AP Ethernet parameters, use the **capwap ap ethernet** command.

capwap ap ethernet tag *ethernet-vlan-id*

Syntax Description	<i>ethernet-vlan-id</i> Ethernet VLAN ID; valid range is 0 to 4094. If you enter the VLAN ID value as 0, the VLAN tagging is disabled.
---------------------------	--

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure Ethernet VLAN tagging on the AP:

```
cisco-ap# capwap ap ethernet tag 2
```

capwap ap hostname

To configure AP hostname, use the **capwap ap hostname** command.

capwap ap hostname *ap-name*

Syntax Description

ap-name AP
name

Command Modes

Privileged EXEC (#)

Usage Guidelines

If the AP is already associated with a Cisco WLC, the new hostname is reflected on the Cisco WLC only after the AP dissociates and reassociates with the Cisco WLC.

Command History

Release Modification

8.1.111.0 This command was
introduced.

Examples

The following example shows how to configure a hostname for the AP:

```
cisco-ap# capwap ap hostname cisco-wave2-ap-2802
```

capwap ap ip

To configure static IP address and DNS for the CAPWAP AP, use the **capwap ap ip** command.

```
capwap ap ip static-ip-addr static-netmask ip-addr-default-gateway [ip-addr-dns1 | ip-addr-dns2]
[domain-name]
```

Syntax Description		
<i>static-ip-addr</i>		Static IP address of the AP
<i>static-netmask</i>		Static netmask
<i>ip-addr-default-gateway</i>		IP address of the default gateway
[<i>ip-addr-dns1</i> <i>ip-addr-dns2</i>]	(Optional parameters)	IP address(es) of the DNS
[<i>domain-name</i>]	(Optional parameter)	Domain name

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure static IP address and DNS for the CAPWAP AP:

```
cisco-ap# capwap ap ip 209.165.200.225 255.255.255.224 209.165.200.227 209.165.200.226
example.org
```

capwap ap lag

To configure CAPWAP lag, use the **capwap ap lag** command.

```
capwap ap lag {enable | disable}
```

Syntax Description

enable Enables LAG

disable Disables LAG

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable LAG on the AP:

```
cisco-ap# capwap ap lag enable
```

capwap ap mesh strict-wired-uplink

To configure the root access points (RAPs) to stay as persistent RAPs even if the wired uplink is lost, use the **capwap ap mesh strict-wired-uplink** command.

capwap ap mesh strict-wired-uplink {enable | disable}

Syntax Description

enable Enables strict wired uplink on the Cisco AP.

disable Disables strict wired uplink on the Cisco AP.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.9	This command was introduced.
Cisco IOS XE Gibraltar 16.11.1	

Examples

The following example shows how to enable the root access points (RAPs) to stay as persistent RAPs even if the wired uplink is lost:

```
cisco-ap# capwap ap mesh strict-wired-uplink enable
```

capwap ap mode

To configure AP mode, use the **capwap ap mode** command.

```
capwap ap mode { bridge | local }
```

Syntax Description

bridge Enables bridge mode

local Enables local mode

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to configure the AP to operate in local mode:

```
cisco-ap# capwap ap mode local
```

capwap ap restart

To restart the CAPWAP protocol, use the **capwap ap restart** command.

capwap ap restart

Syntax Description	restart Restart the CAPWAP protocol
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to restart CAPWAP protocol:

```
cisco-ap# capwap ap restart
```




clear Commands

- [clear avc nbar](#), on page 22
- [clear counters](#), on page 23
- [clear cts](#), on page 24
- [clear datapath](#), on page 25
- [clear dot11](#) , on page 26
- [clear logging](#), on page 27

clear avc nbar

To clear AVC NBAR statistics, use the **clear avc nbar** command.

clear avc nbar statistics

Syntax Description	statistics Clears AVC NBAR statistics
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to clear AVC NBAR statistics:

```
cisco-ap# clear avc nbar statistics
```

clear counters

To clear 802.11 radio statistics, use the **clear counters** command.

clear counters **Dot11Radio** *interface-number* | **client** | **fast-path** **profinet** | **wired** *interface-number* **MIB-stats**

Syntax Description	
Dot11Radio	(Optional) Clears the Dot11 interface statistics.
<i>interface-number</i>	Dot11Radio interface number; valid value is 0 or 1.
client	Clears the client statistics.
fast-path	Clears the controller fast-path statistics.
profinet	Clears the profinet statistics.
wired	Clears the wired interface statistics.
<i>interface-number</i>	Wired interface number, valid value is between 0 and 3.
MIB-stats	Clears the AP Internal-Switch MIB counters.

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.
	8.7	This command was enhanced by adding client , fast-path , profinet , wired parameters.

Examples

The following example shows how to clear 802.11 interface statistics for the interface number specified:

```
cisco-ap# clear counters Dot11Radio 1
```

clear cts

To clear the statistics of Cisco TrustSec Security, use the **clear cts** command.

clear cts role-based counters [**all** | **client** *mac-addr* | **from** *sgt* **to** *dgt*]

Syntax Description

counters	Clears Cisco TrustSec summary counters
all	Clears all Cisco TrustSec counters
client <i>mac-addr</i>	Clears the Cisco TrustSec counters for a client MAC address specified in xx:xx:xx:xx:xx:xx format
from	Specifies the source group tag for filtered traffic
<i>sgt</i>	Security Group Tag (SGT); valid values are 0 to 65535
to	Specifies the destination group tag for filtered traffic
<i>dgt</i>	Destination Group Tag (DGT); valid values are 0 to 65535

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

This example shows you how to clear all the statistics of Cisco TrustSec Security counters:

```
cisco-ap# clear cts role-based counters all
```

clear datapath

To clear the datapath counters or drops, use the **clear datapath** command.

```
clear datapath {drops | statistics}
```

Syntax Description

drops Clears the datapath drop counters

statistics Clears the datapath counters

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

This example shows you how to clear the datapath drop counters:

```
cisco-ap# clear datapath drops
```

clear dot11

To clear the 802.11 configuration, use the **clear dot11** command.

clear dot11 sensor

Syntax Description	sensor Clears the sensor configuration and reboots
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

This example shows you how to clear the 802.11 configuration:

```
cisco-ap# clear dot11 sensor
```

clear logging

To clear the logging details, use the **clear logging** command.

```
clear logging [capwap | message | warning]
```

Syntax Description	
capwap	(Optional) Clears CAPWAP logging details
message	(Optional) Clears message logging details
warning	(Optional) Clears warnings logging details

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

This example shows you how to clear the CAPWAP logging details:

```
cisco-ap# clear logging capwap
```




config Commands

- [config ap address](#) , on page 30
- [config ap client-trace](#), on page 31
- [config ap client-trace filter](#), on page 33
- [config ap client-trace output](#), on page 34
- [config boot baudrate](#), on page 35
- [config boot break](#), on page 36
- [config boot crashkernel](#), on page 37
- [config boot debug-memory](#), on page 38
- [config boot manual](#), on page 39
- [config boot path](#), on page 40
- [config cts debug enforcement host_ip](#), on page 41
- [config cts debug enforcement rate](#), on page 42
- [config cts debug enforcement permissions](#), on page 43
- [config cts debug enforcement protocol](#), on page 44

config ap address

To configure the AP IPv4 or IPv6 address, use the **config ap address** command.

```
config ap address ipv4 { dhcp | static { static-ip-addr static-netmask default-gateway-ip-addr | ipv6
{ auto-config {enable | disable} | dhcp | disable | link-local ipv6-addr | static ipv6-addr ipv6-prefix
gateway-ipv6-addr
```

Syntax Description		
	ipv4	Configure IPv4 address
	ipv6	Configure IPv6 address
	auto-config	Auto configure IPv6 address
	dhcp	Configure IPv6 DHCP
	auto-config	
	auto-config	

Command Default None.

Command History	Release	Modification
		This command was introduced.

Usage Guidelines

Examples

Related Commands	Command	Description

config ap client-trace

To configure client trace on the access point, use the **config ap client-trace** command.

```
config ap client-trace {address {add | clear-all | delete} | all-clients {enable | disable} | filter {all
{enable | disable} | arp {enable | disable} | assoc {enable | disable} | auth {enable | disable} | dhcp
{enable | disable} | eap {enable | disable} | icmp {enable | disable} | ndp {enable | disable} | probe
{enable | disable}} | inline-mon {enable | disable} | output console-log | start | stop}
```

Syntax Description

addresses	Configure clients to trace. Specify the MAC address of the client
add	Specifies a client to trace
clear-all	Delete all client traces on this access point
delete	Deletes client address to be traced. Takes a client MAC address
all-clients	Trace all clients
enable	Enables trace for all clients
disable	Disables trace for all clients
filter	Sets filters for client tracing
all	Traces all filters
arp	Traces ARP packets Use the enable or disable keyword to enable or disable this filter.
assoc	Traces ASSOC packets
auth	Traces auth packets
dhcp	Traces DHCP packets
eap	Traces EAP packets
icmp	Traces ICMP packets
ndp	Traces NDP packets
probe	Trace probe packets.
inline-mon	Enables or disables inline monitoring
output	Enables or disables logging to the console or log file
<i>console-log</i>	Specifies console log keyword
start	Starts client tracing
stop	Stops client tracking

Command Modes Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to start client tracing on the AP:

```
cisco-ap# config ap client-trace start
```

config ap client-trace filter

To set filters for client trace, use the **config ap client-trace filter** command.

```
config ap client-trace filter { all [ disable | enable ] | arp [ disable | enable ] |
assoc [ disable | enable ] | auth [ disable | enable ] | dhcp [ disable | enable ] |
eap [ disable | enable ] | icmp [ disable | enable ] | ndp [ disable | enable ] }
```

Syntax Description	
all	Trace all filters
arp	Trace ARP packets
assoc	Trace ASSOC packets
auth	Trace auth packets
dhcp	Trace DHCP packets
eap	Trace EAP packets
icmp	Trace ICMP packets
ndp	Trace NDP Packets

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

To set filters for client trace, use this command:

```
cisco-ap# config ap client-trace filter
```

config ap client-trace output

To configure the trace output, use the **config ap client-trace output** command.

```
config ap client-trace output console-log {disable | enable}
```

Syntax Description	
console-log	Displays trace output to console and log
disable	Disables trace output to console and log
enable	Enables trace output to console and log

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows you how to configure the trace output:

```
cisco-ap# config ap client-trace output
```

config boot baudrate

To set the baud rate, use the **config boot baudrate** command.

config boot baudrate {115200 | 9600}

Syntax Description	<hr/> <i>115200</i> Sets the baud rate to 115200 <hr/> <i>9600</i> Sets the baud rate to 9600 <hr/>				
Command Default	The default config boot baud rate is 9600.				
Command Modes	Privileged EXEC (#)				
Command History	<table><thead><tr><th>Release</th><th>Modification</th></tr></thead><tbody><tr><td>8.1.111.0</td><td>This command was introduced.</td></tr></tbody></table> <hr/>	Release	Modification	8.1.111.0	This command was introduced.
Release	Modification				
8.1.111.0	This command was introduced.				

Examples

The following example shows how to configure the baud rate to 9600:

```
cisco-ap# config boot baudrate 9600
```

config boot break

To enable break, use the **config boot break** command.

config boot break {**enable** | **disable**}

Syntax Description

enable Enables boot break

disable Disables boot break

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable boot break:

```
cisco-ap# config boot break enable
```


config boot crashkernel

To enable or disable kernel crash, use the **config boot crashkernel** command.

```
config boot crashkernel {enable | disable}
```

Syntax Description	enable Enables kernel crash
---------------------------	------------------------------------

	disable Disables kernel crash
--	--------------------------------------

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable kernel crash:

```
cisco-ap# config boot crashkernel enable
```

config boot debug-memory

To enable memory debug, use the **config boot debug-memory** command.

config boot debug-memory {enable | disable}

Syntax Description	
enable	Enables memory debug
disable	Disables memory debug

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

This example shows you how to enable memory debug:

```
cisco-ap# config boot debug-memory enable
```

config boot manual

To enable manual boot of the AP, use the **config boot manual** command.

```
config boot manual {enable | disable}
```

Syntax Description	enable Enables manual boot
	disable Disables manual boot

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to enable manual boot:

```
cisco-ap# config boot manual enable
```

config boot path

To configure the boot path, use the **config boot path** command.

config boot path {1 | 2}

Syntax Description	{1 2} Path to be specified as Part 1 or Part 2
---------------------------	--

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure the booth path as 1:

```
cisco-ap# config boot path 1
```

config cts debug enforcement host_ip

To filter the SGACL enforcement debugs based on the host IP, use the **config cts debug enforcement host_ip** command.

```
config cts debug enforcement host_ip {ipv4 dst-ip [src-ip] | ipv6 dst-ip [src-ip] }
```

Syntax Description

ipv4 *dst-ip* [*src-ip*] Displays only the IPv4 SGACL enforcement debugs based on the destination and, optionally, source IP addresses

ipv6 *dst-ip* [*src-ip*] Displays only the IPv6 SGACL enforcement debugs based on the destination and, optionally, source IP addresses

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows you how to filter the IPv4 SGACL enforcement debugs based on the host IP:

```
cisco-ap# config cts debug enforcement host_ip ipv4 209.165.200.224 209.165.200.227
```

config cts debug enforcement rate

To configure the rate of printing of debug logs, use the **config cts debug enforcement rate** command.

```
config cts debug enforcement rate {X Y }
```

Command Modes

Privileged EXEC (#)

Syntax Description

rate Configure the rate of printing debug logs

X Number of packets whose debugs are to be displayed for every *Y* number of packets processed; valid range is between 0 to 10000

Y Number of packets to be processed; valid range is between 0 to 10000

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to configure the rate of printing of debug logs such that debugs of 100 packets are displayed for every 500 packets processed:

```
cisco-ap# config cts debug enforcement rate 100 500
```

config cts debug enforcement permissions

To filter SGACL enforcement debugs based on source group tag (SGT) and destination group tag (DGT), use the **config cts debug enforcement permissions** command.

```
config cts debug enforcement permissions { dgt | sgt } tag-id
```

Syntax Description

dgt Destination group tag

sgt Source group tag

tag-id Tag identifier; valid values are between 0 to 65535

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows you how to filter SGACL enforcement debugs for a destination group tag whose ID is 600:

```
cisco-ap# config cts debug enforcement permissions dgt 600
```

config cts debug enforcement protocol

To filter SGACL enforcement debugs based on protocol, use the **config cts debug enforcement protocol** command.

```
config cts debug enforcement protocol {protocol-id | icmp | tcp | udp}
```

Syntax Description

protocol-id Protocol ID; valid values are between 0 to 65535

icmp Filter SGACL enforcement for ICMP traffic

tcp Filter SGACL enforcement for TCP traffic

udp Filter SGACL enforcement for UDP traffic

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows you how to filter SGACL enforcement debugs based on protocol for UDP traffic:

```
cisco-ap# config cts debug enforcement protocol udp
```




debug Commands

- [debug arp](#), on page 47
- [debug ble](#), on page 48
- [debug capwap client](#), on page 49
- [debug capwap client avc](#), on page 50
- [debug cdp](#), on page 51
- [debug cleanair](#), on page 52
- [debug dhcp](#), on page 53
- [debug dot11](#), on page 54
- [debug dot11 client datapath](#), on page 55
- [debug dot11 client level](#), on page 56
- [debug dot11 driver slot](#), on page 57
- [debug dot11 firmware](#), on page 59
- [debug dot11 sensor](#), on page 60
- [debug dtls client](#), on page 61
- [debug ethernet](#), on page 62
- [debug flexconnect](#), on page 63
- [debug lldp](#), on page 64
- [debug memory](#), on page 65
- [debug memory pool](#), on page 66
- [debug memory pool alloc](#), on page 67
- [debug memory pool free](#), on page 68
- [debug mesh](#), on page 69
- [debug mesh adjacency](#), on page 70
- [debug mesh path-control](#), on page 71
- [debug rrm neighbor](#), on page 72
- [debug rrm reports](#), on page 73
- [debug sip](#), on page 74
- [debug wips](#), on page 75
- [debug authentication interface](#), on page 76
- [debug process memory](#), on page 77
- [debug traffic](#), on page 78
- [debug tunnel](#), on page 79
- [debug client trace](#), on page 80

- [no](#), on page 81
- [traceroute](#), on page 82
- [undebug](#), on page 83

debug arp

To enable debugging of ARP, use the **debug arp** command.

debug arp { **errors** | **events** | **packets** }

Syntax Description	errors Enable debugging of ARP errors
	events Enable debugging of ARP events
	packets Enable debugging of ARP Tx and Rx packets
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of ARP errors:

```
cisco-ap# debug arp errors
```

debug ble

To enable debugging of Bluetooth Low Energy (BLE), use the **debug ble** command.

debug ble {critical | error | events | fastpath {rssi | scan | sync} | receive | transmit}

Syntax Description

critical	Enables debugging of BLE critical events
error	Enables debugging of BLE error events
events	Enables debugging of BLE events
fastpath {rssi scan sync}	Shows data exported to CMX. The following options are available: <ul style="list-style-type: none"> • RSSI data • Scan data • Sync data
receive	Enables debugging of BLE packet received from BLE radio
transmit	Enables debugging of BLE packet transmitted to BLE radio

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.7	This command was introduced.

Examples

The following example shows how to enable debugging of BLE critical events:

```
cisco-ap# debug ble critical
```

debug capwap client

To enable debugging of CAPWAP clients, use the **debug capwap client** command.

debug capwap client { **ble** | **detail** | **efficient-upgrade** | **error** | **events** | **flexconnect** | **info** | **keepalive** | **payload** | **pmtu** | **qos** | **reassembly** | **security** }

Syntax Description	ble	detail	efficient-upgrade	error	events	flexconnect	info	keepalive	payload	pmtu	qos	reassembly	security
	Enables debugging of CAPWAP BLE detail	Enables debugging of CAPWAP detail	Enables debugging of image predownload	Enables debugging of CAPWAP error	Enables debugging of CAPWAP events	Enables debugging of CAPWAP FlexConnect mode event	Enables debugging of CAPWAP information	Enables debugging of CAPWAP keepalive	Enables debugging of CAPWAP payload	Enables debugging of CAPWAP path MTU	Enables debugging of CAPWAP QoS	Enables debugging of CAPWAP reassembly	Enables debugging of CAPWAP security

Command Modes Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of CAPWAP client detail:

```
cisco-ap# debug capwap client detail
```

debug capwap client avc

To enable debugging of CAPWAP client AVC, use the **debug capwap client avc** command.

debug capwap client avc { **all** | **detail** | **error** | **event** | **info** | **netflow** { **all** | **detail** | **error** | **event** | **packet** } | **numflows** }

Syntax Description

all	Enables debugging of all CAPWAP client AVC
detail	Enables debugging of CAPWAP AVC detail
error	Enables debugging of CAPWAP AVC error
event	Enables debugging of CAPWAP AVC event
info	Enables debugging of CAPWAP AVC information
netflow	Enables debugging of CAPWAP client AVC NetFlow
netflow all	Enables debugging of all CAPWAP client AVC NetFlow
netflow detail	Enables debugging of CAPWAP client AVC NetFlow detail
netflow error	Enables debugging of CAPWAP client AVC NetFlow error
netflow event	Enables debugging of CAPWAP client AVC NetFlow event
netflow packet	Enables debugging of CAPWAP client AVC NetFlow packet
numflows	Enables debugging of CAPWAP client AVC numflows

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of all CAPWAP client AVC:

```
cisco-ap# debug capwap client avc all
```

debug cdp

To enable debugging of controller discovery protocol (CDP), use the **debug cdp** command.

debug cdp { **adjacency** | **events** | **ilp** | **packets** }

Syntax Description	
adjacency	Enables debugging of CDP neighbors
events	Enables debugging of CDP events
ilp	Enables debugging of inline power
packets	Enables debugging of CDP packets

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of CDP events:

```
cisco-ap# debug cdp events
```

debug cleanair

To configure debugging of CleanAir, use the **debug cleanair** command.

debug cleanair {bringup | event | logdebuglow | major | nsi | offchan {0 | 1}}

Syntax Description

bringup	Enables debugging of CleanAir port or bringups
events	Enables debugging of normal CleanAir events
logdebug	Logs CleanAir debug output to a logfile
low	Enables debugging of hex dump of some messages
major	Enables debugging of major CleanAir events
nsi	Enables debugging of NSI messages
offchan 0 1	Enables debugging of CleanAir MSMT requests. You have to specify the radio slot as either 0 or 1

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of major CleanAir events:

```
cisco-ap# debug cleanair major
```


debug dhcp

To configure debugging of DHCP, use the **debug dhcp** command.

debug dhcp {**errors** | **events** | **packets**}

Syntax Description	
errors	Enables debugging of DHCP errors
events	Enables debugging of DHCP events
packets	Enables debugging of DHCP packets

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of DHCP errors:

```
cisco-ap# debug dhcp errors
```

debug dot11

To enable debugging of 802.11, use the **debug dot11** command.

debug dot11 { **critical** | **errors** | **events** | **info** }

Syntax Description	
critical	Enables 802.11 critical level debugging
errors	Enables 802.11 error level debugging
events	Enables 802.11 event level debugging
info	Enables 802.11 information level debugging

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of 802.11 error level:

```
cisco-ap# debug dot11 errors
```

debug dot11 client datapath

To enable debugging of 802.11 client datapath, use the **debug dot11 client datapath** command.

```
debug dot11 client datapath {{arp | dhcp | eapol} {addr {mac-addr1 | mac-addr2 | mac-addr3
| mac-addr4} | all}} | dns-acl}
```

Syntax Description		
arp		Enables client datapath ARP debugging
dhcp		Enables client datapath DHCP debugging
eapol		Enables client datapath EAPOL debugging
{ addr all }		Option to specify MAC address of specific clients or all clients
{ <i>mac-addr1</i> <i>mac-addr2</i> <i>mac-addr3</i> <i>mac-addr4</i> }		MAC addresses of clients that you have to enter
dns-acl		Enables client datapath DNS-ACL debugging

Command Modes Privileged EXEC (#)

Command History **Release** **Modification**

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of client datapath ARP:

```
cisco-ap# debug dot11 client datapath arp
```

debug dot11 client level

To enable 802.11 client debugging level, use the **debug dot11 client level** command.

```
debug dot11 client level {critical | errors | events | info} {addr {mac-addr1 | mac-addr2 |
mac-addr3 | mac-addr4} | all}
```

Syntax Description		
critical		Enables client critical level debugging
errors		Enables client error level debugging
events		Enables client event level debugging
info		Enables client information level debugging
{addr all}		Option to specify MAC address of specific clients or all clients
	<i>{mac-addr1 mac-addr2 mac-addr3 mac-addr4}</i>	MAC addresses of clients that you have to enter

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of all clients at the event level:

```
cisco-ap# debug dot11 client level events all
```

debug dot11 driver slot

To enable debugging of 802.11 drivers, use the **debug dot11 driver slot** command.

```
debug dot11 driver slot {0 | 1} {all | {cac {info | metrics}} | chd | save-acnt-data |
save-on-failure [extended] | stop-on-failure | tsm | vim | type { all | assoc | auth |
dhcp | eap | icmp | probe } mac-addr1 | mac-addr2 | mac-addr3 | mac-addr4
```

Syntax Description

slot {0 1}	Enables 802.11 driver debugs per radio
all	Enables all 802.11 driver debugs
cac	Enables 802.11 CAC debugs
cac info	Enables 802.11 CAC info level debugs
cac metrics	Enables debugging of 802.11 CAC metrics
chd	Enables 802.11 CHD debugs
save-acnt-data	Saves the radio accounting data
save-on-failure	Saves the radio crash information upon radio failure
save-on-failure extended	Saves extended information on radio failure
stop-on-failure	Stops the AP from reboot on radio failure
tsm	Enables 802.11 traffic stream metric debugs
vim	Enables 802.11 video metric debugs
type	Enables the debug types.
all	Enables the all type debugging.
assoc	Enables the assoc debugging.
auth	Enables the auth debugging.
dhcp	Enables the dhcp debugging.
eap	Enables the eap debugging.
icmp	Enables the icmp debugging.
probe	Enables the probe debugging.
<i>mac-addr</i>	MAC addresses of the clients. You can enter upto four MAC addresses.

Command Modes

Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.
	8.5.140.0 and 8.8	This command was enhanced by adding the type parameter.

Examples

The following example shows how to enable debugging of CAC at the information level:

```
cisco-ap# debug dot11 driver slot cac info
```

debug dot11 firmware

To debug the 802.11 firmware, use the **debug dot11 firmware** command.

```
debug dot11 firmware slot slot_ID level { all | critical | emergency | error | info }
addressmac-addr1 | mac-addr2 | mac-addr3 | mac-addr4
```

Syntax Description

<i>slot_ID</i>	Enables 802.11 driver debugs per radio
all	Enables all the debug levels.
critical	Enables critical level debugs.
emergency	Enables emergency level debugs.
error	Enables error level debugs.
info	Enables info level debugs.
address	To add client address for driver/firmware debugging.
<i>mac-addr</i>	MAC addresses of the clients. You can enter upto four MAC addresses.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.5.140.0 and 8.8	This command was introduced.

Example

The following example shows how to enable debugging of 802.11 emergency level:

```
cisco-wave2-ap# debug dot11 firmware slot 1 emergency address 92:FB:D6:B3:7A:6C
```

debug dot11 sensor

To enable debugging of 802.11 sensors, use the **debug dot11 sensor** command.

```
debug dot11 sensor { dns | file-transfer | mail-server | ping | radius | ssh | telnet | web-server }
```

Syntax Description

dns	Enables debugging of 802.11 sensor DNS
file-transfer	Enables debugging of 802.11 sensor file transfer
mail-server	Enables debugging of 802.11 sensor mail server
ping	Enables debugging of 802.11 sensor ping
radius	Enables debugging of 802.11 sensor radius
ssh	Enables debugging of 802.11 sensor SSH
telnet	Enables debugging of 802.11 sensor Telnet.
web-server	Enables debugging of 802.11 sensor web server

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of 802.11 sensor file transfer:

```
cisco-ap# debug dot11 sensor file-transfer
```


debug dtls client

To configure DTLS client error and event debugging, use the **debug dtls client** command.

```
debug dtls client {error | event [detail]}
```

Syntax Description	error	Configures debugging of DTLS client errors
	event [detail]	Configures debugging of DTLS client events

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of DTLS client events:

```
cisco-ap# debug dtls client event
```

debug ethernet

To configure Ethernet debugging, use the **debug ethernet** command.

debug ethernet *interface-number* { **both** | **rcv** | **xmt** }

Syntax Description

<i>interface-number</i>	Interface number that you have to enter as either 0 or 1
both	Enables debugging of both transmission and reception
rcv	Enables debugging of reception
xmt	Enables debugging of transmission

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of transmission for interface 0:

```
cisco-ap# debug ethernet 0 xmt
```

debug flexconnect

To debug FlexConnect features, use the **debug flexconnect** command.

```
debug flexconnect {acl | cckm | dot11r | event | multicast {igmp | traffic} | pmk | proxy-arp | vsa | wlan-vlan | wsastats}
```

Syntax	Description
acl	Configures debugging of FlexConnect ACL
cckm	Configures debugging of CCKM
dot11r	Configures debugging of 802.11r
event	Configures debugging of wireless control protocol (WCP) events
multicast igmp	Configures debugging of Multicast IGMP
multicast traffic	Configures debugging of Multicast traffic
pmk	Configures debugging of opportunistic key caching (OKC) or pairwise master key caching
vsa	Configures debugging of AAA vendor specific attributes (VSA)
wlan-vlan	Configures debugging of WLAN-VLAN mapping
wsastats	Configures debugging of RADIUS or DHCP wireless service assurance statistics

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of FlexConnect ACL:

```
cisco-ap# debug flexconnect acl
```

debug lldp

To debug LLDP, use the **debug lldp** command.

debug lldp { **errors** | **events** | **packet** }

Syntax Description	
errors	Debugs LLDP errors
events	Debugs LLDP events
packet	Debugs LLDP packets

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of LLDP errors:

```
cisco-ap# debug lldp errors
```

debug memory

To debug memory, use the **debug memory** command.

```
debug memory {clear | save}
```

Syntax Description

clear Removes memory debug upon boot-up

save Saves current debug level and applies it upon following boots

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to remove memory debug upon boot-up:

```
cisco-ap# debug memory clear
```

debug memory pool

To debug memory pool, use the **debug memory pool** command.

```
debug memory pool { diff | realtime interval 1-1000000-seconds | start }
```

Syntax Description		
diff		Shows memory pool debug difference in detail
realtime interval <i>1-1000000-seconds</i>		Configures realtime interval for the memory pool
start		Starts the debug for the memory pool

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure realtime interval of 180 seconds for the memory pool:

```
cisco-ap# debug memory pool realtime interval 180
```

debug memory pool alloc

To debug memory pool allocation calls, use the **debug memory pool alloc** command.

```
debug memory pool alloc {all | name pool-name} {diff | realtime interval 1-1000000-seconds | start}
```

Syntax Description		
all		Configures debug for all memory pool allocation calls
name <i>pool-name</i>		Configures debug for a specific memory pool's allocation call
diff		Shows memory pool debug allocation call difference in detail
realtime interval <i>1-1000000-seconds</i>		Configures realtime interval for the memory pool allocation calls
start		Starts the debug for the memory pool allocation calls

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to configure the start of the debug for all memory pool allocation calls:

```
cisco-ap# debug memory pool alloc all start
```

debug memory pool free

To debug memory pool free calls, use the **debug memory pool free** command.

```
debug memory pool free {all | name pool-name} {diff | realtime interval 1-1000000-seconds | start}
```

Syntax Description

all	Configures debug for all memory pool free calls
name <i>pool-name</i>	Configures debug for a specific memory pool's free call
diff	Shows memory pool debug free call difference in detail
realtime interval <i>1-1000000-seconds</i>	Configures realtime interval for the memory pool free calls
start	Starts the debug for the memory pool free calls

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to configure the start of the debugging of all memory pool free calls:

```
cisco-ap# debug memory pool free all start
```


debug mesh

To configure debugging of mesh networks, use the **debug mesh** command.

debug mesh {**channel** | **clear** | **convergence** | **events** | **forward-mcast** | **forward-packet** | **forward-table** | **linktest** | **path-control** | **port-control** | **security** | **trace**}

Syntax Description

channel	Configures debugging of mesh channel
clear	Resets all mesh debugs
convergence	Configures debugging of mesh convergence
events	Configures debugging of mesh events
forward-mcast	Configures debugging of mesh forwarding Multicast
forward-packet	Configures debugging of mesh forwarding packets
forward-table	Configures debugging of mesh forwarding table
linktest	Configures debugging of mesh linktest
port-control	Configures debugging of mesh port control
security	Configures debugging of mesh security
trace	Configures debugging of mesh trace

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of mesh channel:

```
cisco-ap# debug mesh channel
```

debug mesh adjacency

To debug mesh adjacency, use the **debug mesh adjacency** command.

debug mesh adjacency { **child** | **clear** | **dfs** | **message** | **packet** | **parent** }

Syntax Description	
adjacency	Debug mesh adjacency
child	Debug mesh adjacency child
clear	Debug clear mesh adjacency
dfs	Debug mesh DFS
message	Debug mesh adjacency messages
packet	Debug mesh adjacency packet
parent	Debug mesh adjacency parent

Command Modes Privileged EXEC (#)

Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of mesh adjacency parent:

```
cisco-ap# debug mesh adjacency parent
```

debug mesh path-control

To configure debugging of mesh path control, use the **debug mesh path-control** command.

debug mesh path-control {**error** | **events** | **packets** }

Syntax Description	
error	Configures debugging of mesh path control errors
events	Configures debugging of mesh path control events
packets	Configures debugging of mesh path control packets

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of mesh path control errors:

```
cisco-ap# debug mesh path-control error
```

debug rrm neighbor

To enable RRM neighbor debugging, use the **debug rrm neighbor** command.

debug rrm neighbor {**tx** | **rx** | **detail**}

Syntax Description	
tx	Enable RRM neighbor Tx debugging
rx	Enable RRM neighbor Rx debugging
detail	Enable RRM neighbor detail debugging

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of RRM neighbor transmissions:

```
cisco-ap# debug rrm neighbor tx
```

debug rrm reports

To enable RRM reports debugging, use the **debug rrm reports** command.

debug rrm reports

Syntax Description	reports Enables RRM report debugging
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of RRM reports:

```
cisco-ap# debug rrm reports
```

debug sip

To enable session initiation protocol (SIP) debugging, use the **debug sip** command.

debug sip { **all** | **tx** | **rx** }

Syntax Description	all Enabling SIP transmission and reception debugging
	tx Enabling SIP transmission debugging
	rx Enabling SIP reception debugging
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to enable debugging of SIP transmissions and reception:

```
cisco-ap# debug sip all
```

debug wips

To enable wIPS debugging, use the **debug wips** command.

debug wips {**errors** | **events** | **critical**}

Syntax Description

errors Enable wIPS error level debugging

events Enable wIPS event level debugging

critical Enable wIPS critical level debugging

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to enable wIPS error level debugging:

```
cisco-ap# debug wips errors
```

debug authentication interface

To enable FlexConnect radio interface debugging, use the **debug authentication interface** command.

debug authentication interface *interface-name* { **all** | **dot11** | **dot1x** | **driver** | **others** | **radius** | **wpa** }

Syntax Description

<i>interface-name</i>	Name of the interface to debug
all	Enable all parts debugging
dot11	Enable 802.11 module debugging
dot1x	Enable 802.1x module debugging
driver	Enable driver module debugging
others	Enable other non-module parts debugging
radius	Enable RADIUS module debugging
wpa	Enable WPA module debugging

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of WPA module:

```
cisco-ap# debug authentication interface management wpa
```


debug process memory

To process memory debugging, use the **debug process memory** command.

```
debug process memory {diff | realtime [interval interval-in-seconds ] | start}
```

Syntax Description

diff	Process memory debug show diff
realtime	Process memory real time debug
<i>interval</i>	Update interval; valid range 1 to 1000000 seconds
start	Process memory debug start

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable the start of debugging of process memory:

```
cisco-ap# debug process memory start
```

debug traffic

To enable traffic debugging, use the **debug traffic** command.

```
debug traffic {host {icmpv6 | ip | ipv6 | tcp | udp { verbose}} | wired {ip | tcp | udp {verbose}}}
```

Syntax Description

host	Enabling host traffic debugging
wired	Enabling wired traffic debugging
verbose	Display verbose output
icmpv6	Enabling host ICMPv6 traffic dump
ip	Enabling host IP traffic dump
ipv6	Enabling host IPv6 traffic dump
tcp	Enabling TCP traffic dump
udp	Enabling UDP traffic dump

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable debugging of host IP traffic dump:

```
cisco-ap# debug traffic host ip
```

debug tunnel

To configure debugging of tunnel, use the **debug tunnel** command.

debug tunnel eogre

Syntax Description	eogre Configures debugging of EoGRE tunnel				
Command Modes	Privileged EXEC (#)				
Command History	<table><thead><tr><th>Release</th><th>Modification</th></tr></thead><tbody><tr><td>8.1.111.0</td><td>This command was introduced.</td></tr></tbody></table>	Release	Modification	8.1.111.0	This command was introduced.
Release	Modification				
8.1.111.0	This command was introduced.				

Examples

The following example shows how to enable debugging of EoGRE tunnel:

```
cisco-ap# debug tunnel eogre
```

debug client trace

To enable client trace debugging, use the **debug client trace** command.

```
debug client trace {all | address mac-address | enable | filter {assoc | auth | dhcp | eap | icmp | mgmt | probe | proto}}
```

Syntax Description		
	all	Configure all clients tracing
	address	Configure address(es) to trace
	<i>mac-address</i>	MAC address to trace
	enable	Enable tracing
	filter	Configure trace filter
	assoc	Trace Association packets
	auth	Trace Authentication packets
	dhcp	Trace DHCP packets
	eap	Trace EAP packets
	icmp	Trace ICMP packets
	mgmt	Trace probe, assoc, auth, EAP packets
	probe	Trace probe packets
	proto	Trace DHCP, ICMP packets

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to enable tracing of all clients:

```
cisco-ap# debug client trace all
```

no

To negate a command or set to its defaults, use the **no** command.

no

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

To negate a command or set to its defaults, use this command:

```
cisco-ap# no debug
```

tracert

To view the routes followed by packets traveling in the network, use the **tracert** command.

tracert *destination-address*

Syntax Description	<i>destination-address</i> IP address of the destination of the packets				
Command Modes	Privileged EXEC (#)				
Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>8.1.111.0</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	8.1.111.0	This command was introduced.
Release	Modification				
8.1.111.0	This command was introduced.				

Examples

The following example shows how to view the routes followed by packets traveling in the network, with a destination IP address specified:

```
cisco-ap# tracert 209.165.200.224
```

undebug

To disable debugging on the access point, use the **undebug** command.

undebug [**all**]

Syntax Description

a Disables all debugging messages.

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to disable all debugging messages:

```
cisco-ap# undebug all
```




show Commands

- [show ap client-trace status](#), on page 87
- [show arp](#), on page 88
- [show avc cft](#), on page 89
- [show avc nbar](#), on page 90
- [show avc netflow flows](#), on page 91
- [show avc status](#), on page 92
- [show boot](#), on page 93
- [show capwap](#), on page 94
- [show capwap client](#), on page 95
- [show capwap client trace](#), on page 96
- [show capwap ids sig](#), on page 97
- [show cdp](#), on page 98
- [show class-map](#), on page 99
- [show cleanair debug](#), on page 100
- [show client statistics](#), on page 101
- [show clock](#), on page 102
- [show configuration](#), on page 103
- [show controller ble](#), on page 104
- [show controllers dot11Radio](#), on page 105
- [show controllers nss status](#), on page 106
- [show controllers wired](#), on page 107
- [show crypto](#), on page 108
- [show debug](#), on page 109
- [show dhcp](#), on page 110
- [show dot11 qos](#), on page 111
- [show filesystems](#), on page 112
- [show flash](#), on page 113
- [show flexconnect](#), on page 114
- [show flexconnect ocap firewall](#), on page 115
- [show flexconnect wlan](#), on page 116
- [show interfaces dot11Radio](#), on page 117
- [show interfaces network](#), on page 118
- [show interfaces wired](#), on page 119

- [show inventory](#), on page 120
- [show ip](#), on page 121
- [show lacp](#), on page 122
- [show logging](#), on page 123
- [show memory](#), on page 124
- [show policy-map](#), on page 126
- [show processes](#), on page 127
- [show processes memory](#), on page 128
- [show rrm](#), on page 130
- [show rrm rogue containment](#), on page 131
- [show rrm rogue detection](#), on page 132
- [show running-config](#), on page 134
- [show spectrum](#), on page 136
- [show tech-support](#), on page 137
- [show version](#), on page 138
- [show trace dot11_chn](#), on page 139
- [show trace](#), on page 140
- [show wips](#), on page 141

show ap client-trace status

To view the AP client trace details, use the **show ap client-trace status** command.

```
show ap client-trace { events { all | mac word | system } | skb { drop-list | stats } | status }
```

Syntax Description	
events	View client trace event information
all	Displays all client trace events
system	Displays all system events
mac	Displays client trace events for specific MAC address
<i>word</i>	Specific client MAC address
skb	Displays client trace SKB information
drop-list	Displays client trace SKB drop list information
stats	Displays client trace SKB statistics
status	Displays client trace configuration

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the AP client trace status:

```
cisco-ap# show ap client-trace status
```

show arp

To view the ARP table, use the **show arp** command.

show arp

Syntax Description	arp Shows ARP table
Command Modes	User EXEC (>) Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

The following example shows a sample output of the command:

```
cisco-ap# show arp

Address Age (min)      Hardware Addr
 9.11.8.1             0 84:80:2D:A0:D2:E6
9.11.32.111          0 3C:77:E6:02:33:3F
```

show avc cft

To view the AVC client flow table information, use the **show avc cft** command.

show avc cft *word*

Syntax Description	<i>word</i> Client MAC address
Command Modes	User EXEC (>) Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

The following example shows how to view the AVC client flow table:

```
cisco-ap# show avc cft 02:35:2E:03:E0:F2
```

show avc nbar

To view the AVC NBAR information, use the **show avc nbar** command.

```
show avc nbar {statistics | build | version}
```

Syntax Description

statistics Displays NBAR build details

build Displays NBAR statistics

version Displays NBAR and PP version

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the AVC NBAR build information:

```
cisco-ap# show avc nbar build
```

show avc netflow flows

To list all the flows currently cached and to be sent to the Cisco WLC, use the **show avc netflow flows** command.

```
show avc netflow flows {download | upload}
```

Syntax Description	
download	Lists currently cached download flows
upload	Lists currently cached upload flows

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view all the currently cached flows:

```
cisco-ap# show avc netflow flows
```

show avc status

To list the AVC provisioning status per WLAN/VAP, use the **show avc status** command.

show avc status

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view AVC provisioning status per WLAN/VAP:

```
cisco-ap# show avc status

VAP FNF-STATUS AVC-QOS-STATUS
 0 Disabled Disabled
 1 Disabled Disabled
 2 Disabled Disabled
 3 Disabled Disabled
 4 Disabled Disabled
 5 Disabled Disabled
 6 Disabled Disabled
 7 Disabled Disabled
 8 Disabled Disabled
 9 Disabled Disabled
10 Disabled Disabled
11 Disabled Disabled
12 Disabled Disabled
13 Disabled Disabled
14 Disabled Disabled
15 Disabled Disabled
```


show boot

To show boot attributes, use the **show boot** command.

show boot

Command Modes	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view boot attributes:

```
cisco-ap# show boot

BOOT path-list:      part2
Console Baudrate:   9600
Enable Break:       yes
Manual Boot:        no
Memory Debug:       no
Crashkernel:
```

show capwap

To display CAPWAP options, use the **show capwap** command.

```
show capwap [{ip | mcast | traffic}]
```

Syntax Description	
client	CAPWAP client information
ids	CAPWAP ID information
ip	CAPWAP IP configuration
location	CAPWAP location information
mcast	CAPWAP multicast information
pnp	PNP information
traffic	CAPWAP traffic information

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the CAPWAP multicast information:

```
cisco-ap# show capwap mcast
```

show capwap client

To display CAPWAP client information, use the **show capwap client** command.

```
show capwap client {callinfo info | detailrcb | rcb | config | ha | msginfo | timers | traffic}
```

Syntax Description	
callinfo <i>info</i>	CAPWAP client call information
detailrcb	CAPWAP client detailed RCB information
rcb	CAPWAP client RCB information
config	CAPWAP client config information
ha	CAPWAP client HA parameters
msginfo	CAPWAP client messages information
timers	CAPWAP client timers
traffic	CAPWAP client 802.11 traffic information

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view CAPWAP client traffic information:

```
cisco-ap# show capwap client traffic
```

show capwap client trace

To display CAPWAP trace, use the **show capwap client trace** command.

show capwap client trace {clear | delete | disable | save | start | stop}

Syntax Description	
clear	Clears trace
delete	Deletes trace
disable	Disables trace at boot
enable	Enables trace at boot
save	Saves trace
start	Starts trace
stop	Stops trace

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view CAPWAP client trace:

```
cisco-ap# show capwap client trace
```

show capwap ids sig

To display CAPWAP ID signatures, use the **show capwap ids sig** command.

```
show capwap ids sig [{list | stats}]
```

Syntax Description	list	Signature list entries
	stats	Signature attack statistics
Command Modes	User EXEC (>) Privileged EXEC (#)	
Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example show how to view CAPWAP ID signature statistics:

```
cisco-ap# show capwap ids sig stats
```

show cdp

To display CDP options, use the **show cdp** command.

show cdp {**entry device** *device-name* | **inline_power** | **interface** | **neighbors** | **traffic**}

Syntax Description

entry device <i>device-name</i>	Information for specific neighbor entry whose name you must enter
inline_power	Inline power negotiation information
interface	CDP interface status and configuration
neighbors	CDP neighbor entries
traffic	CDP statistics

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view information for a specific neighbor entry:

```
cisco-ap# show cdp entry device mydevice
```

show class-map

To display CPL class map, use the **show class-map** command.

show class-map

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view CPL class map:

```
cisco-ap# show class-map
```

show cleanair debug

To display cleanair debug settings, use the **show cleanair debug** command.

show cleanair debug

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view CleanAir debug settings:

```
cisco-ap# show cleanair debug
```


show client statistics

To display client statistics, use the **show client statistics** command.

```
show client statistics client-mac-address
```

Syntax Description	<i>client-mac-address</i> MAC address of the client
---------------------------	---

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view client statistics:

```
cisco-ap# show client statistics 70:DB:98:66:34:FA
```

show clock

To display the system clock, use the **show clock** command.

show clock

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the system clock:

```
cisco-ap# show clock
```

show configuration

To display the contents of the non-volatile memory, use the **show configuration** command.

show configuration rlan

Command Modes

Privileged EXEC (#)

Syntax Description

Command History

Release	Modification
8.1.111.0	This command was introduced.
8.9	This command was enhanced by adding rlan parameter.
8.10.112.0	The output of this command was enhanced to show the status of broken antenna detection.

The following example shows how to view the AP configuration details:

```
cisco-ap# show configuration

AP Name           : AP58AC.78DC.C2F0
Admin State       : Enabled
AP Mode           : FlexConnect
AP Submode        : Not Configured
Location          : default location
Reboot Reason     : Reload command
.
.
AP Link LAG status : Disabled
AP WSA Mode       : Enabled
Vlan Interface    : Disabled

Broken antenna detection : Enabled (Global)
RSSI Failure Threshold  : 40
Weak RSSI             : 60
Detection Time        : 12
If any broken antenna? : ALL
AP58AC.78DC.C2F0#
```

show controller ble

To view Bluetooth Low Energy radio interface parameter information, use the **show controller ble** command.

show controller ble *ble-interface-number* { **broadcast** | **counters** | **floor-tag** *floor-beacon-mac-addr* | **interface** | **local** | **scan** { **brief** | **detail** *floor-beacon-mac-addr* } | **timers**}

Syntax Description		
	<i>ble-interface-number</i>	BLE interface number that you must enter; Valid value is 0
	broadcast	Displays BLE broadcast summary information
	counters	Displays BLE transport counters information
	floor-tag <i>floor-beacon-mac-addr</i>	Displays sync data of the floor beacon whose MAC address you must specify
	interface	Displays BLE interface summary information
	local	Displays sync information of host BLE radio
	scan brief	Displays brief BLE scan summary information
	scan detail <i>floor-beacon-mac-addr</i>	Displays BLE scan summary information in detail; you must specify the floor beacon MAC address
	timers	Displays BLE timers information

Command Modes Privileged EXEC (#)

Command History

Release Modification

8.7 This command was introduced.

Examples

To view the BLE timers information, use this command:

```
cisco-ap# show controller ble 0 timers
Timers
-----
Scan timer status      : Running
Scan timer interval    : 10 secs
Scan started at       : 0D:00H:04M:28S ago
Last scan done at     : 0D:00H:00M:06S ago
```

If scanning is working as expected, the 'Last scan done at' time should always be less than or equal to the scan interval set.

show controllers dot11Radio

To display dot11 interface information, use the **show controllers dot11Radio** command.

```
show controllers dot11Radio dot11-interface-no{antenna | { atfconfiguration | statistics } | bandselect
| client { client-mac-addr | all detail } | frequency | powercfg | powerreg | radiostats | rate | vlan
| wlan { wlan-id | all detail } }
```

Syntax	Description
<i>dot11-interface-no</i>	Dot11Radio interface number.
atf configuration	Displays the AirTime Fairness configuration.
atf statistics	Displays the AirTime Fairness statistics.
bandselect	Displays the bandselect statistics.
antenna	Displays the antenna settings
client <i>client-mac-addr</i>	Displays the details of the client whose MAC address is specified.
detail	Displays the TID statistics for all the clients.
frequency	Displays the frequency information.
powercfg	Displays the configured power information.
powerreg	Displays the transmit power information.
radio-stats	Displays the radio statistics.
rate	Displays the rate information.
vlan	Displays the VLAN summary.
wlan <i>wlan-id</i>	Displays the VLAN/WLAN details of the WLAN ID specified.
detail	Displays the TID statistics for all the clients.

Command Modes User EXEC (>)

Command History

Release Modification

8.1.111.0 This command was introduced.

8.9 This command was enhanced by adding the **bandselect** ,**client all detail** , **wlan** parameters.

The following example shows how to view 802.11 interface information for interface number 1:

```
cisco-ap# show controllers dot11Radio 1
```

show controllers nss status

To display NSS information, use the **show controllers nss status** command.

show controllers nss status

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was
introduced.

The following example shows how to view NSS information:

```
cisco-ap# show controllers nss status
```

show controllers wired

To view the wired interface, use the **show controllers wired** command.

show controllers wired *wired-interface-number*

Syntax Description	<i>wired-interface-number</i> Wired interface number from 0 to 3				
Command Modes	Privileged EXEC (#)				
Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>8.1.111.0</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	8.1.111.0	This command was introduced.
Release	Modification				
8.1.111.0	This command was introduced.				

The following example shows how to view information about the controllers' wired interface whose ID is 1:

```
cisco-ap# show controllers wired 1

wired1    Link encap:Ethernet  HWaddr C8:8B:6A:33:59 eMac Status: DOWN
          inet addr:9.11.8.104  Bcast:9.255.255.255  Mask:255.255.255.255
          DOWN BROADCAST RUNNING PROMISC MULTICAST  MTU:2400  Metric:1
          RX packets:38600 errors:0 dropped:1 overruns:0 frame:0
          TX packets:179018 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:80
          RX bytes:3812643 (3.6 MiB)  TX bytes:54721869 (52.1 MiB)

Gig Emacl Counters
-----
0 Good octets rx, 0 Bad octets rx, 0 Unicast frames rx,
0 Broadcast frames rx, 0 Multicast frames rx, 0 64 byte frames rx,
0 65_TO_127 byte frames, 0 128_TO_255 byte frames, 0 256_TO_511 byte frames,
0 512_TO_1023 byte frames, 0 1024_TO_MAX byte frames, 0 Good octets tx,
0 Unicast frames tx, 0 Multicast frames tx, 0 Broadcast frames tx,
0 Crc errors sent, 0 Flow control rx, 0 Flow control tx,
0 Rx fifo overrun, 0 Undersized rx, 0 Fragments rx,
0 Oversize rx, 0 Jabber rx, 0 Mac rx error,
0 Bad crc event, 0 Collision, 0 Late collision,
```

show crypto

To view the crypto attributes, use the **show crypto** command.

show crypto

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was
 introduced.

The following example shows how to view the crypto attributes:

```
cisco-ap# show crypto
```


show debug

To view the debugs enabled, use the **show debug** command.

show debug

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the debugs that are in enabled state:

```
cisco-ap# show debug
```

show dhcp

To view the status of Dynamic Host Configuration Protocol (DHCP), use the **show dhcp** command.

```
show dhcp {lease | servers}
```

Syntax Description	
lease	Displays the DHCP addresses leased from a server
servers	Displays the known DHCP servers

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release Modification
	8.1.111.0 This command was introduced.

The following example shows how to view the status of DHCP addresses leased from a server:

```
cisco-ap# show dhcp lease
```

show dot11 qos

To view the Quality of Service (QoS) parameters for 802.11 network, use the **show dot11 qos** command.

show dot11 qos

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the Quality of Service (QoS) parameters for 802.11 network:

```
cisco-ap# show dot11 qos
```

show filesystems

To view the filesystem information, use the **show filesystems** command.

show filesystems

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the filesystem information:

```
cisco-ap# show filesystems
```

Filesystem	Size	Used	Available	Use%	Mounted on
/dev/ubivol/storage	57.5M	1.9M	52.6M	4%	/storage

show flash

To view the flash contents, use the **show flash** command.

```
show flash [{cores [detail core-file-name ] | crash | syslogs}]
```

Syntax Description		
	cores	Displays the core files in flash
	detail	Displays the core file contents
	<i>core-file-name</i>	The core file name
	crash	Displays the crash files in flash
	syslogs	Displays the syslogs files in flash

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the details of a core file in flash:

```
cisco-ap# show flash cores detail filename1
```

show flexconnect

To view the flexconnect information for an access point, use the **show flexconnect** command.

```
show flexconnect {calea | cckm | client [aaa-override | counter | priority] | dot11r |
mcast | oeap | pmk | status | vlan-acl | wlan}
```

Syntax	Description
calea	Displays the calea information
cckm	Displays the CCKM cache entry information
client	Displays the client information
aaa-override	Specifies the AAA override parameters
counter	Specifies the counter for all clients
priority	Specifies the client priority
dot11r	Displays the 802.11r cache entry information
mcast	Displays the multicast information
oeap	Displays the FlexConnect OEAP information
pmk	Displays the OKC or PMK cache entry information
status	Displays the standalone status
vlan-acl	Displays the VLAN ACL mapping
wlan	Displays the WLAN configuration

Command Modes
User EXEC (>)
Privileged EXEC (#)

Command History	Release Modification
	8.1.111.0 This command was introduced.

The following example shows how to view the information about a client of a FlexConnect AP:

```
cisco-ap# show flexconnect client
```

show flexconnect oeap firewall

To view the OEAP firewall information, use the **show flexconnect oeap firewall** command.

```
show flexconnect oeap firewall [{dmz | filtering | forwarding}]
```

Syntax Description

dmz	Displays the OEAP firewall DMZ information
filtering	Displays the OEAP firewall filtering information
forwarding	Displays the OEAP firewall port forwarding information

Command Modes

User EXEC (>)
Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the OEAP firewall DMZ information:

```
cisco-ap# show flexconnect oeap firewall dmz
```

show flexconnect wlan

To view the WLAN configuration for Flexconnect AP mode, use the **show flexconnect wlan** command.

```
show flexconnect wlan [{l2acl | qos | vlan}]
```

Syntax Description	
l2acl	Specifies the Layer 2 ACL mapping for WLAN
qos	Specifies the QoS parameters for WLAN
vlan	Specifies the VLAN mapping for WLAN

Command Modes	
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the WLAN Layer 2 ACL mapping for the Flexconnect AP:

```
cisco-ap# show flexconnect wlan l2acl
```


show interfaces dot11Radio

To view the interface status and configuration for an 802.11 radio, use the **show interfaces dot11Radio** command.

```
show interfaces dot11Radio radio-interface-number {dfs | memory [memory-address length | firmware] | mumimo wlan-number | sniffer | statistics | wlanwlan-id datapathcounters | statistics }
```

Syntax	Description
<i>radio-interface-number</i>	Specifies the interface number for 802.11 radio. The valid range is from 0 to 1
dfs	Displays the DFS statistics
memory	Displays the dump radio memory
<i>memory-address</i>	Specifies the memory address. The valid range is between 0 and ffffffff
<i>length</i>	Specifies the length. The valid range is from 0 to 64
firmware	Dumps firmware logs
mumimo	Displays the multiuser MIMO statistics information
<i>wlan-number</i>	The 802.11-specific value whose valid range is from 0 to 15.
sniffer	Displays the sniffer mode statistics
statistics	Displays the statistics information for 802.11 radio
wlan <i>wlan-id</i>	Displays the specified WLAN information
datapath	Displays the datapath counters.
counters	Displays the datapath counters and drops.

Command Modes Privileged EXEC (#)

Command History **Release** **Modification**

8.1.111.0 This command was introduced.

8.9 This command was enhanced by adding the **datapath** parameter.

The following example shows how to view the DFS statistics for a 802.11 interface whose number is 1:

```
cisco-ap# show interfaces dot11Radio 1 dfs
```

```
DFS Data:
Radar Detected:          0
Inactive Radar Detected: 0
```

show interfaces network

To view the Linux network interfaces, use the **show interfaces network** command.

show interfaces network

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the Linux network interfaces:

```
cisco-ap# show interfaces network
```

show interfaces wired

To view the wired interface, use the **show interfaces wired** command.

show interfaces wired *wired-interface-number* {**MIB-stats** | **datapath counters**}

Syntax Description	
<i>wired-interface-number</i>	Wired interface number; valid range is between 0 to 3
MIB-stats	Displays the AP internal-Switch MIB counters.
datapath	Displays the datapath counters.
counters	Displays the datapath counters and drops.

Command Modes Privileged EXEC (#)

Command History **Release** **Modification**

8.1.111.0 This command was introduced.

8.9 This command was enhanced by adding the **datapath** parameter.

The following example shows how to view the wired interface whose number is 1:

```
cisco-ap# show interfaces wired 1
```

show inventory

To view the physical inventory, use the **show inventory** command.

show inventory

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the physical inventory:

```
cisco-ap# show inventory
```

```
NAME: AP2800, DESCR: Cisco Aironet 2800 Series (IEEE 802.11ac) Access Point  
PID: AIR-AP2802I-D-K9 , VID: V01, SN: XXXXXXXXXXXX
```

show ip

To view the IP information, use the **show ip** command.

```
show ip {access-lists | interface brief | route | tunnel [eogre {domain | forwarding-table
| gateway} | fabric | summary | sip-snooping { stats | status} ] }
```

Syntax	Description
access-lists	Lists the IP access lists
interface	Displays the IP interface status and configuration
brief	Displays the brief summary of IP status and configuration
route	Displays the IP routing table
tunnel	Displays the IP tunnel information
eogre	Displays the EoGRE tunnel information
domain	Displays the EoGRE tunnel domain information
forwarding-table	Displays the EoGRE tunnel encapsulation and decapsulation information
gateway	Displays the EoGRE tunnel gateway information
fabric	Displays the IP fabric tunnel information
summary	Displays the information for all tunnels
sip-snooping	Displays the SIP snooping options.
stats	Displays the transmitted and received SIP snooping statistics.
status	Displays the SIP snooping status.

Command Modes	Description
	User EXEC (>)
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.
	8.9	This command was enhanced by adding the sip-snooping parameter.

The following example shows how to view information about the lists the IP access lists:

```
cisco-ap# show ip access-lists
```

show lacp

To view the Link Aggregation Control Protocol (LACP) options, use the **show lacp** command.

```
show lacp {counters | internal | neighbors}
```

Syntax Description	
counters	Displays traffic information
internal	Displays internal information
neighbors	Displays LACP neighbor entries

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the LACP traffic information:

```
cisco-ap# show lacp counters
```

show logging

To view the contents of logging buffers, use the **show logging** command.

show logging

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the contents of logging buffers:

```
cisco-ap# show logging
```

show memory

To display memory usage on an access point, use the **show memory** command.

show memory [{**detail** | **pool** | **summary**}]

Syntax Description	detail	Displays detailed system memory usage
	pool	Displays system memory pool
	summary	Display system memory usage statistics

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the system memory usage statistics:

```
cisco-ap# show memory
Memory summary:
MemTotal:      1030608 kB
MemFree:       713832 kB
MemAvailable:  710492 kB
Buffers:       0 kB
Cached:        88224 kB
SwapCached:   0 kB
Active:        28932 kB
Inactive:      82872 kB
Active(anon):  28900 kB
Inactive(anon): 82812 kB
Active(file):  32 kB
Inactive(file): 60 kB
Unevictable:   0 kB
Mlocked:       0 kB
SwapTotal:     0 kB
SwapFree:      0 kB
Dirty:         0 kB
Writeback:     0 kB
AnonPages:    23580 kB
Mapped:        11380 kB
Shmem:         88132 kB
Slab:          132140 kB
SReclaimable: 3368 kB
SUnreclaim:   128772 kB
KernelStack:  864 kB
PageTables:    748 kB
NFS_Unstable: 0 kB
Bounce:        0 kB
WritebackTmp: 0 kB
CommitLimit:  515304 kB
Committed_AS: 193960 kB
VmallocTotal: 1024000 kB
VmallocUsed:   69808 kB
VmallocChunk: 915324 kB
```



```
System Memory:
      total      used      free      shared      buffers
Mem:    1030608    316848    713760         0         0
-/+ buffers:    316848    713760
Swap:         0         0         0
```

show policy-map

To view policy maps on access point, use the **show policy-map** command.

show policy-map

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the policy maps on the access point:

```
cisco-apshow policy-map
```

show processes

To view process utilization details, use the **show processes** command.

```
showprocesses {cpu cpu-number | dmalloc {capwap | wcp} | status}
```

Syntax Description

cpu <i>cpu-number</i>	Displays the specified CPU's utilization of the processes; valid range of values for the CPU number is between 0 to 3
dmalloc	Displays the process utilization of the dmalloc processes
capwap	Displays dmalloc statistics for CAPWAP
wcp	Displays dmalloc statistics for WCP
status	Displays watchdog process status

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the process watchdog status:

```
cisco-ap# show processes status
      Process           Alive           Monitored
      capwap            True            True
      switchdrvr        True            False
      wcpd               True            True
      kclick             True            True
      cleanaird          True            True
      mrvlfwd            True            True
```

show processes memory

To display the processes on the access point, use the **show processes memory** command.

show processes memory {maps | smaps} pid *pid-number*

Syntax Description	maps	Displays maps for the processes
	smaps	Displays smaps for the processes
	pid	Process ID that you have to specify <i>pid-number</i>

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the list of processes utilizing the memory on the access point:

```
cisco-ap# show processes memory

Mem total:1030608 anon:23876 map:11424 free:712728
  slab:132748 buf:0 cache:88284 dirty:0 write:0
Swap total:0 free:0
  PID  VSZ^VSZRW  RSS (SHR)  DIRTY (SHR)  STACK  COMMAND
6227  56500 53464 1168   732 1144   732  132 /usr/sbin/mrvlfd
6283  27536 20668 13032 2400 13032 2400 132 /usr/sbin/capwapd
6297  24880 10612 14536 1376 14536 1376 132 wcpd
6255  9612  6600 1508 1052 1508 1052 132 /usr/sbin/cleanaird
5122  9556  4144 2664 2012 2664 2012 132 /usr/bin/capwap_brain
29097 7148  1536 3560 2392 3556 2388 132 /usr/sbin/cisco_shell
3142  6828  1216 2992 2264 2992 2264 132 /usr/sbin/cisco_shell
5106  4588  404 1912 1644 1912 1644 132 /usr/bin/fastcgi -s /tmp/fcgi_sock
5108  4588  404 1912 1644 1912 1644 132 /usr/bin/slowcgi -s /tmp/slow_fcgi_sock
6084  4544  452  928  360  928  360 132 /usr/sbin/lighttpd -f /etc/lighttpd.conf
6214  3692  344 1420  960 1420  960 132 tamd_proc ap-tam 1 0 -debug err
6213  3556  340 1460 1104 1460 1104 132 tams_proc -debug err
6133  3396  400 1196  976 1196  976 132 /usr/bin/poder_agent
4689  3176  336 1012  812 1012  812 132 /usr/bin/sync_log /storage/syslogs/13
6143  3140  304 1428 1204 1428 1204 132 /usr/bin/failover
4716  3136  284  616  436  616  436 132 watchdogd
6121  3116  280  988  820  988  820 132 bigacl_d
5084  3112  272  952  804  952  804 132 /usr/bin/led_core
6181  1884  320 1044  260 1044  260 132 perl /usr/bin/drt.pl
  1  1596  196  492  412  492  412 132 init
30914 1596  196  428  344  428  344 132 top -m -b -n 1
6145  1596  196  248  176  248  176 132 {S80cisco} /bin/sh /etc/init.d/S80cisco
start
30912 1592  192  424  356  424  356 132 {show_process_me} /bin/ash
/usr/bin/cli_scripts/show_process_memory.sh 0 0 0 0 0 0 0 0 0
30911 1592  192  400  336  400  336 132 /bin/sh -c
/usr/bin/cli_scripts/show_process_memory.sh 0 0 0 0 0 0 0 0 0 | more
```

```
4684 1592 192 368 304 368 304 132 syslogd -S -s 100 -b 1 -L -R 255.255.255.255
30913 1592 192 332 264 332 264 132 more
4688 1584 184 344 284 344 284 132 klogd
4686 1584 184 320 264 320 264 132 printkd
30906 1584 184 284 228 284 228 132 sleep 10
29085 1452 332 640 416 640 416 132 /usr/sbin/dropbear -E -j -k -d
/storage/dropbear/dropbear_dss_host_key -r /storage/dropbear/dropbear_rsa_host_key
6209 1384 264 416 364 416 364 132 /usr/sbin/dropbear -E -j -k -d
/storage/dropbear/dropbear_dss_host_key -r /storage/dropbear/dropbear_rsa_host_key
8411 1096 212 444 336 444 336 132 dnsmasq -C /etc/dnsmasq.host.conf
6115 1096 212 436 340 436 340 132 dnsmasq -C /etc/dnsmasq.vaperr.conf
```

show rrm

To view the Radio Resource Management (RRM) properties, use the **show rrm** command.

```
show rrm {hyperlocation [level-list] | neighbor-list [details] | receive {configuration | statistics}}
```

Syntax Description

hyperlocation <i>level-list</i>	Displays status of Cisco Hyperlocation on the AP
neighbor-list	Displays neighbor-list statistics
receive	Receive signal strength indicator (RSSI) of the AP
rogue	Displays rogue-related information

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Usage Guidelines

The following example shows how to view the level 1 channel scan list in Hyperlocation:

```
cisco-ap# show rrm hyperlocation level1-list
Level-1 List for 2.4GHz Band
=====
Channel   Width      Serving MAC  Max Clients
-----
 
Level-1 List for 5GHz Band
=====
Channel   Width      Serving MAC  Max Clients
-----
```

show rrm rogue containment

To view rogue containment information on an access point, use the **show rrm rogue containment** command.

```
show rrm rogue containment {ignore | info} Dot11Radio radio-interface-number
```

Syntax Description		
ignore		Displays list of rogue APs that are configured to be ignored
info		Displays rogue containment configuration and statistics for an AP
Dot11Radio		Specifies the Dot11Radio interface keyword.
<i>radio-interface-number</i>		Slot of the radio interface; valid values are 0 and 1
Command Modes	Privileged EXEC (#)	
Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the rogue containment and statistics for the 802.11 interface numbered 1:

```
cisco-ap# show rrm rogue containment info Dot11Radio 1
Rogue Containment Info and Stats for slot 1:
bssid client-addr contain-type channels

Request Status count
      Submit      0
      Success     0
      Timeout     0
      Error       0
      Tuned       0
      Flushed     0
      Bad Channel  0
      Tail Dropped 0
      Cancelled   0
NDP DFS Tx Cancelled 0
      Tx Failed   0
      Created     0
```

show rrm rogue detection

To view RRM rogue detection configuration parameters, use the **show rrm rogue detection** command.

show rrm rogue detection {**adhoc** | **ap** | **clients** | **config** | **rx-stats**} **Dot11Radio** *radio-interface-number*

Syntax Description		
adhoc		Displays the primary ad hoc rogue AP list for a 802.11 radio slot; valid values are 0 and 1
ap		Displays rogue detection parameters for the AP for a 802.11 radio slot; valid values are 0 and 1
clients		Displays primary list of rogue clients
config		Displays rogue detection configuration on the AP
rx-stats		Displays rogue detection receive statistics on the 802.11 interfaces of an AP
Dot11Radio		Specifies 802.11 radio interface
<i>radio-interface-number</i>		The 802.11 radio interface number; valid values are 0 and 1

Command Modes Privileged EXEC (#)

Command History **Release Modification**

8.1.111.0 This command was introduced.

The following example shows how to view the RRM rogue detection configuration details:

```
cisco-ap# show rrm rogue detection config

Rogue Detection Configuration for Slot 0:
Rogue Detection Mode : Enabled
Rogue Detection Report Interval : 10
Rogue Detection Minimum Rssi : -90
Rogue Detection Transient Interval : 0
Rogue Detection Flex Contain : Disabled
Rogue Detection Flex Contain Adhoc : Disabled
Rogue Detection Flex Contain SSID : Disabled
Rogue Containment Autorate : Disabled
Scan Duration : 180000
Channel Count : 11
Transient Threshold : 0

Rogue Detection Configuration for Slot 1:
Rogue Detection Mode : Enabled
Rogue Detection Report Interval : 10
Rogue Detection Minimum Rssi : -90
Rogue Detection Transient Interval : 0
Rogue Detection Flex Contain : Disabled
Rogue Detection Flex Contain Adhoc : Disabled
Rogue Detection Flex Contain SSID : Disabled
Rogue Containment Autorate : Disabled
```



```
Scan Duration : 180000  
Channel Count : 25  
Transient Threshold : 0
```

show running-config

To display the contents of the currently running configuration on the access point, use the **show running-config** command.

show running-config

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the contents of the currently running configuration on the access point:

```
cisco-ap# show running-config

AP Name                : ap1540
Admin State            : Enabled
AP Mode                : Local
AP Submode             : None
Location               : default location
Reboot Reason          : Config Mwar
Primary controller name : cisco_3504
Primary controller IP  : <controller-ip-address>
Secondary controller name :
Secondary controller IP :
Tertiary controller name :
Tertiary controller IP  :
Controller from DHCP offer : <controller-dhcp-server-address>
Controller from DNS server : <controller-dns-server-address>
AP join priority        : 1
IP Prefer-mode          : IPv4
CAPWAP UDP-Lite        : Unconfigured
Last Joined Controller name: wlc3504
DTLS Encryption State  : Disabled
Discovery Timer        : 10
Heartbeat Timer        : 30
CDP State              : Enabled
Watchdog monitoring    : Enabled
IOX                    : Disabled
RRM State              : Enabled
LSC State              : Disabled
SSH State              : Enabled
AP Username            : admin
Session Timeout        : 0
Extlog Host            : 0.0.0.0
Extlog Flags           : 0
Extlog Status Interval : 0
Syslog Host            : <syslog-host-ip-address>
Syslog Facility        : 0
Syslog Level           : errors
Core Dump TFTP IP Addr :
Core Dump File Compression : Disabled
Core Dump Filename     :
Client Trace Status    : Enabled(All)
```

```
Client Trace All Clients    : Enabled
Client Trace Filter        : 0x0000000E
Client Trace Out ConsoleLog: Disabled
WLC Link LAG status       : Disabled
AP Link LAG status        : Disabled
AP WSA Mode                : Disabled
```

show spectrum

To view the show commands of the spectrum firmware, use the **show spectrum** command.

show spectrum {list | recover | status }

Syntax Description	list	recover	status
	Lists the spectrum FW data files	Displays the spectrum FW recover count	Displays the spectrum FW status

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the spectrum firmware status:

```
cisco-ap# show spectrum status

Spectrum FW status slot 0:
  version: 1.15.4
  status:  up, crashes 0, resets 0, radio reloads 0
  load:    37.00 34.75 33.50 33.25
  NSI Key: 26c1bd25893a4b6dd3a00fe71735d067
  NSI:     not configured
  reg_wdog: 255 26309 0
  dfs_wdog: 0
  dfs_freq: 0
Spectrum FW status slot 1:
  version: 1.15.4
  status:  up, crashes 0, resets 0, radio reloads 0
  load:    37.25 38.00 38.75 39.00
  NSI Key: 26c1bd25893a4b6dd3a00fe71735d067
  NSI:     not configured
  reg_wdog: 255 26309 0
  dfs_wdog: 0
  dfs_freq: 0
```

show tech-support

To automatically run show commands that display system information, use the **show tech-support** command.

show tech-support

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to automatically run show commands that display system information:

```
cisco-ap# show tech-support
```

show version

To view the software version information of the AP, use the **show version** command.

show version

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the software version information of the AP:

```
cisco-ap# show version
```

show trace dot11_chn

To view off-channel events on 802.11 channel of an AP, use the **show trace dot11_chn** command.

```
show trace dot11_chn {enable | disable | statistics}
```

Syntax Description	enable	disable	statistics
	Enables displaying of off-channel events on the 802.11 radio 0 and 1	Disables displaying of off-channel events on the 802.11 radios 0 and 1	Displays off-channel event statistics on 802.11 radios 0 and 1

Command Modes Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to view off-channel event statistics on 802.11 radios:

```
cisco-ap# show trace dot11_chn statistics

Dot11Radio0 Off-Channel Statistics:
total_count in_prog_count last-chan last-type last-dur
          0             0         0         0         0

Dot11Radio1 Off-Channel Statistics:
total_count in_prog_count last-chan last-type last-dur
          0             0         0         0         0
```

show trace

To view trace logs on the AP, use the **show trace** command.

```
show trace
```

Command Modes	Privileged EXEC (#)
----------------------	---------------------

Command History	Release	Modification
	8.1.111.0	This command was introduced.

The following example shows how to view the trace logs on the AP:

```
cisco-ap# show trace
```


show wips

To view details of the AP that is configured in wIPS mode, use the **show wips** command.

```
show wips {alarm alarm-id | analyzer | buffer | channel channelno | infrastructure-device | neighbors
| node mac mac-address | node number number | object | policy policy-id | policy ssid | session
mac-address | stats | violation node mac-address | violation channel channel-number}
```

Syntax	Description
alarm	Displays statistics of the configured alarm if the AP is configured in wIPS mode; valid values are between 0 and 255
<i>alarm-id</i>	Alarm ID; valid values are between 0 and 255
analyzer	Displays analyzer related statistics
buffer	Displays statistics of the buffer
channel	Displays channel related statistics
<i>channelno</i>	Channel number; valid values are between 0 and 255
infrastructure-device	Displays AP infrastructure information
neighbors	Displays statistics of neighbors.
node	Displays AP node information
mac <i>mac-address</i>	MAC address of the node.
node	Node.
number <i>number</i>	Node number; valid values are between 1 and 500
object	AP object store
policy { <i>policy-id</i> ssid }	AP policy; you must specify either a policy ID or the policy SSID.
session <i>mac-address</i>	Displays node session details; you must enter the MAC address of the node
stats	Displays AP statistics
violation	Tracks AP violations
node <i>mac-address</i>	Tracks node-based violations
channel <i>channel-number alarm-id</i>	Tracks channel-based violations; you must enter channel number and alarm ID

Command Modes Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

The following example shows how to view the wIPS statistics information on the AP:

```
cisco-ap# show wips stats
```



System Management Commands

- [ap-type](#) , on page 144
- [archive](#), on page 145
- [copy](#) , on page 146
- [delete](#), on page 147
- [disable](#), on page 148
- [enable](#), on page 149
- [exec-timeout](#) , on page 150
- [logging](#), on page 151
- [more](#), on page 152
- [reload](#), on page 153
- [terminal](#), on page 154

ap-type

To configure the AP type for an AP, use the **ap-type** command.

ap-type {**capwap** | **mobility-express** *word* | **workgroup-bridge**}

Syntax Description	capwap	Enable the AP as CAPWAP AP type
	mobility-express	Enable the AP as Mobility Express AP type
	<i>word</i>	Enter the TFTP transfer command details in following format: tftp://<tftp-server-ip-address>/<filename with path from root>
	workgroup-bridge	Enable the Workgroup Bridge(WGB) AP type
Command Modes	Privileged EXEC (#)	
Command History	Release Modification	
	8.1.111.0 This command was introduced.	
	8.8.120.0 This command was enhanced by added workgroup-bridge parameter.	

Examples

The following example shows how to configure the AP type to CAPWAP:

```
cisco-ap# ap-type capwap
```

archive

To download the AP image, use the **archive** command.

archive download-sw {/no-reload | /reload | capwap *word*}

Syntax Description	
download-sw	Software download commands
/no-reload	No-reload after loading the image
/reload	Reload after loading the image
capwap	Download the image from the Cisco WLC
<i>word</i>	Enter the image details in the ap image type ap3g3/ap1g4 format
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

copy

To copy a file, use the **copy** command.

copy {**cores** *filename* [**scp:** *scp-url* | **tftp:** *tftp-url*] | **flash** *filename* [**scp:** *scp-url* | **tftp:** *tftp-url*] | **support-bundle** [**scp:** *scp-url* | **tftp:** *tftp-url*] | **syslogs** [*filename* {**scp:** *scp-url* | **tftp:** *tftp-url*}] | **scp:** *scp-url* | **tftp:** *tftp-url*}

Syntax Description

cores	Applies the action on a core file
<i>filename</i>	Name of the file
scp:	Uses the SCP protocol
<i>scp-url</i>	Enter the SCP URL in the following format: username@A.B.C.D:[/dir]/filename
tftp:	Uses the TFTP protocol
<i>tftp-url</i>	Enter the TFTP URL in the following format: A.B.C.D[/dir]/filename
flash	Applies the action on a flash file
support-bundle	Copies the support bundle to the server
syslogs	Applies the action on the syslog file

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

delete

To delete a file, use the **delete** command.

```
delete{/force | /recursive | /rf } cores filename
```

Syntax Description	
/force	Force delete
/recursive	Recursive delete
/rf	Recursive force delete
cores	Apply action on a core file
<i>filename</i>	Filename to delete

Command Modes	
	Privileged EXEC (#)

Command History	Release	Modification
	8.1.111.0	This command was introduced.

Examples

The following example shows how to delete a file:

```
cisco-ap# delete /rf cores file-name
```

disable

To turn off privileged commands, use the **disable** command.

disable

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to turn off privileged commands:

```
cisco-ap# disable
```


enable

To turn on privileged commands, use the **enable** command.

enable

Command Modes

User EXEC (>)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to turn on privileged commands:

```
cisco-ap> enable
```

exec-timeout

To set the exec-timeout, use the **exec-timeout** command.

exec-timeout *timeout-value*

Syntax Description	<i>timeout-value</i> Timeout value; valid values range between 0 to 2147483647
Command Modes	Privileged EXEC (#)
Command History	Release Modification
	8.1.111.0 This command was introduced.

Examples

The following example shows how to set the exec-timeout to 20 seconds:

```
cisco-ap# exec-timeout 20
```

logging

To log commands, use the **logging** command.

logging { **console** [**disable**] | **host** { **clear** | **disable** | **enable** } }

Syntax Description

console Console logging

host Configure syslog server

disable Disable syslog host logging

enable Enable syslog server

clear Clear syslog server IP

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
8.1.111.0	This command was introduced.

Examples

The following example shows how to enable console logging:

```
cisco-ap# logging console
```

more

To display a file, use the **more** command.

more {**flash** | **syslog**} *file-name*

Syntax Description

flash Apply action on a flash file

syslog Apply action on syslog file

name File name

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to display a syslog file named test-log:

```
cisco-ap# more syslog test-log
```

reload

To halt the access point or perform a reboot, use the **reload** command.

reload [{**at** *hours minutes day-of-month year* | **cancel** | **in** *minutes* | **reason** *reason-string*}]

Syntax Description

at Reload the AP at a specific date and time

This keyword takes the hour, minute, day of the month, month, and year as parameters; valid values for the keywords are as follows:

- *hour*: 0 to 23
- *minutes*: 0 to 59
- *day-of-the-month*: 1 to 31
- *month*: 1 to 12
- *year*: 2015-2099

cancel Cancels the pending reload

in Reload after a time interval, which you should specify in terms of minutes; valid values are between 1 to 1440 minutes

reason A string specifying the reason for the reload

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to reload the AP in 10 minutes:

```
cisco-ap# reload in 10
```

terminal

To configure terminal parameters, use the **terminal** command.

terminal {**length** | **monitor** [**disable**] | **type** *word* | **width** *no-of-characters*}

Syntax Description

length Specifies the number of lines on the screen. Valid values are between 0 to 512. Enter 0 if you do not want the outputs to pause.

monitor Specifies the debug output to the current terminal line. Press the enter key to enable monitoring. To disable monitoring, enter the keyword **disable**.

type Specifies the terminal type

width Specifies the width of the display terminal; valid values are between 0 to 132

Command Modes

Privileged EXEC (#)

Command History

Release Modification

8.1.111.0 This command was introduced.

Examples

The following example shows how to configure the terminal length to 50 lines:

```
cisco-ap# terminal length 50
```