在9800 WLC上設定外部Web驗證並疑難排解

目錄

簡介

本檔案介紹如何在Catalyst 9800無線LAN控制器(WLC)上設定和疑難排解外部Web驗證(EWA)。

必要條件

本檔案假設Web伺服器已正確設定為允許外部通訊，且網頁已正確設定為傳送WLC驗證使用者並將使用者端作業階段移至RUN狀態所需的所有引數。

注意：由於外部資源訪問受WLC透過訪問清單許可權的限制，因此網頁中使用的所有指令碼、字型、映像等都需要下載並保留在Web伺服器的本地位置。

使用者身份驗證的必要引數包括：

- buttonClick：需要將此引數設定為值「4」，WLC才能將操作檢測為身份驗證嘗試。
- redirectUrl：當身份驗證成功時，控制器使用此引數中的值將客戶端定向到特定網站。
- err_flag：此引數用於指示某些錯誤，如資訊不完整或身份證明不正確。在成功的身份驗證中

，此引數設定為「0」。
- username：此引數僅用於webauth引數對映，如果將引數對映設定為同意，則可以忽略它。它必須填寫無線客戶端使用者名稱。
- password：此引數僅用於webauth引數對映，如果將引數對映設定為同意，則可以忽略它。它必須填寫無線客戶端密碼。

需求

思科建議您瞭解以下主題：

- 超文字標籤語言(HTML) Web開發
- Cisco IOS®-XE無線功能
- Web瀏覽器開發人員工具

採用元件

本文中的資訊係根據以下軟體和硬體版本：

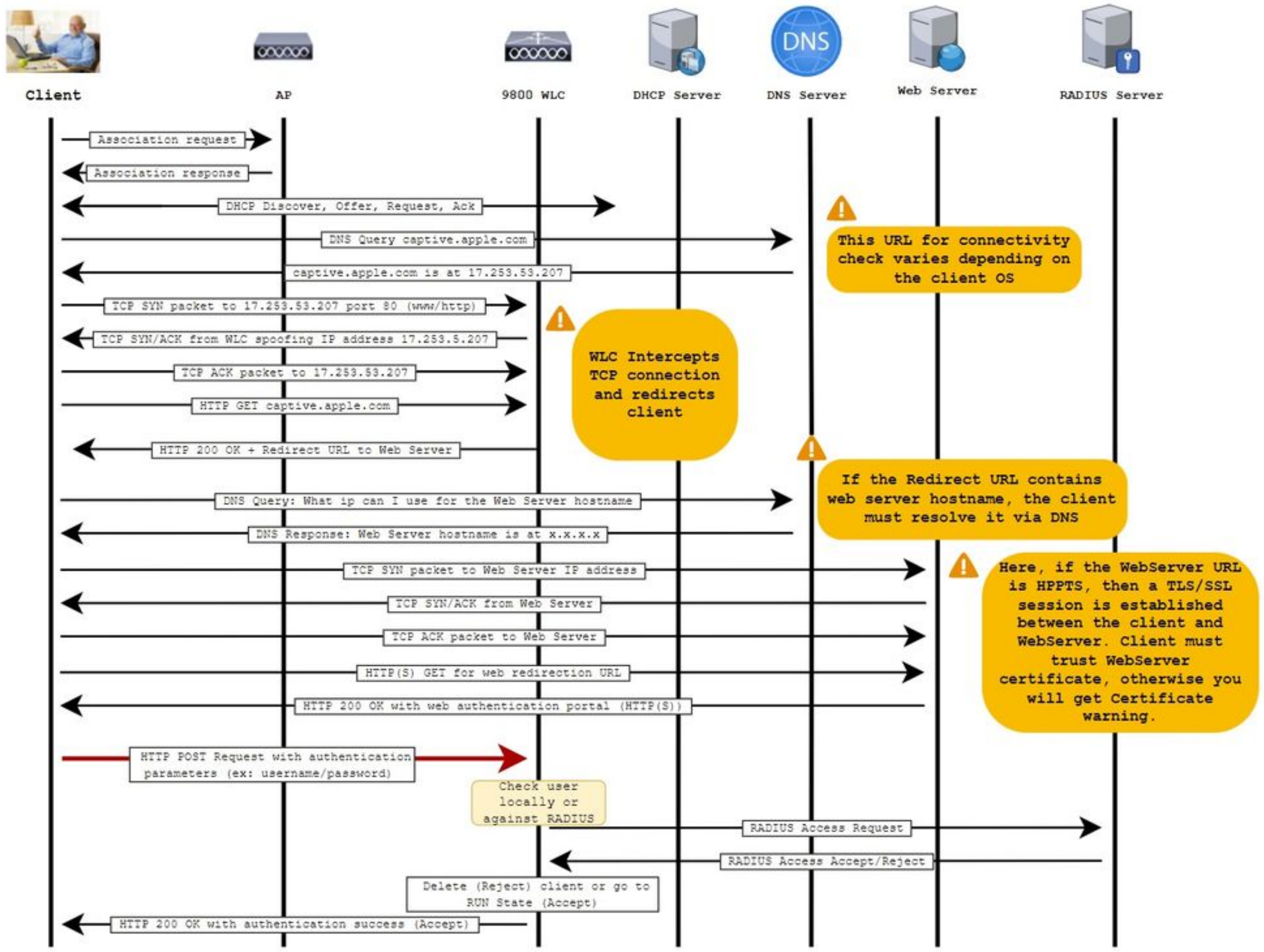
- C9800-CL WLC Cisco IOS®-XE版本17.3.3
- 具有Internet Information Services (IIS)功能的Microsoft Windows Server 2012
- 2802和9117無線存取點

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

背景資訊

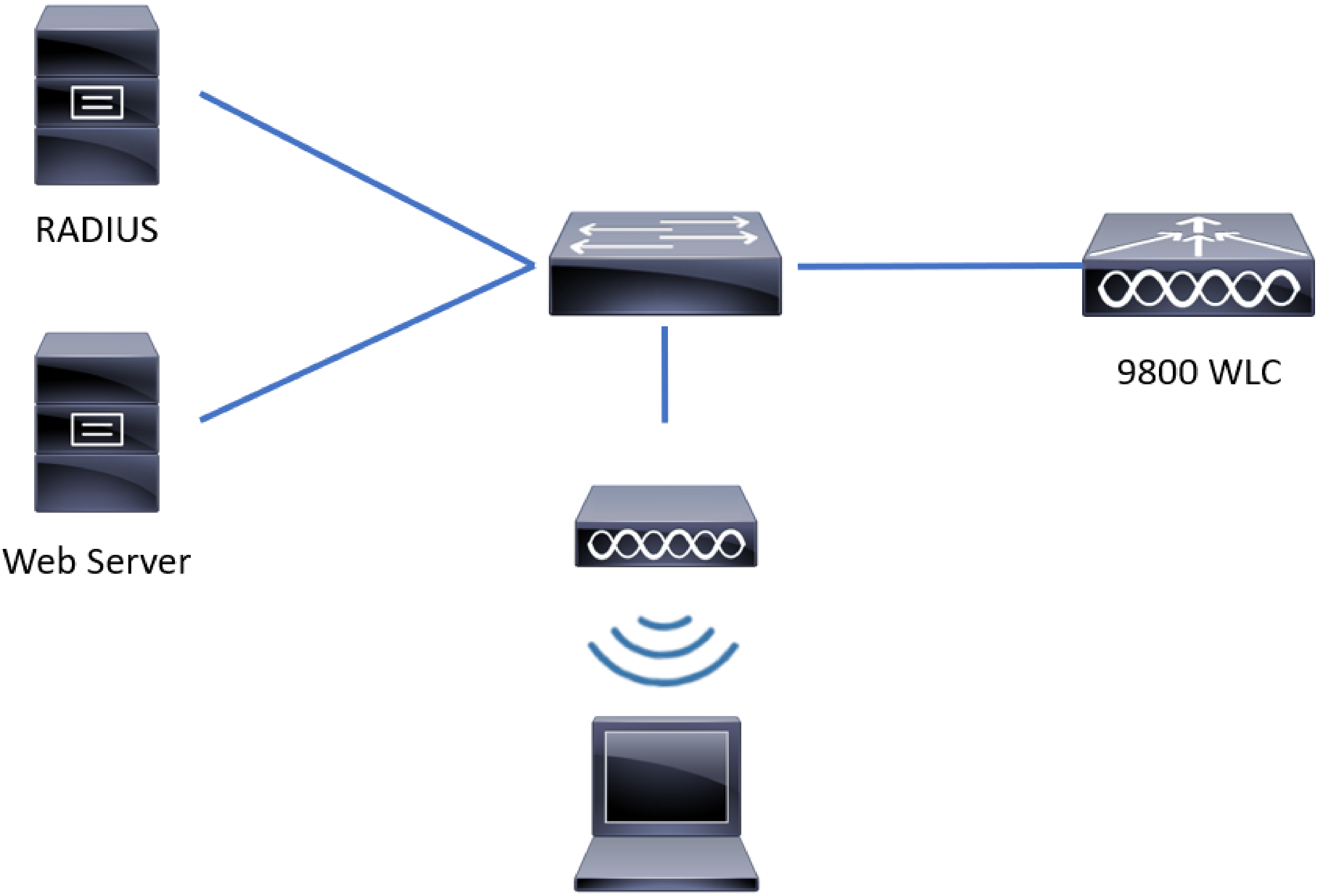
外部Web驗證利用WLC外部裝載的Web入口網站於專用Web伺服器或多用途伺服器(例如身分辨識服務引擎(ISE))，可允許精細存取和管理Web元件。成功將客戶端加入外部Web身份驗證WLAN所涉及的握手在映像中呈現。此影像會列出無線使用者端、WLC、解析統一資源位置(URL)的網域名稱系統(DNS)伺服器，以及WLC在本機驗證使用者憑證的Web伺服器之間的連續互動。此工作流程有助於排除任何故障情況。

注意：在從客戶端到WLC的HTTP POST呼叫之前，如果在引數對映中啟用了安全Web身份驗證，並且如果WLC沒有由受信任的證書頒發機構簽署的信任點，則會在瀏覽器中顯示安全警報。使用者端需要略過此警告，並接受表單重新提交，以便控制器將使用者端工作階段置於RUN狀態。


Client
AP
9800 WLC
DHCP Server
DNS Server
Web Server
RADIUS Server
Association request
Association response
DHCP Discover, Offer, Request, Ack
DNS Query captive.apple.com
captive.apple.com is at 17.253.53.207
TCP SYN packet to 17.253.53.207 port 80 (www/http)
TCP SYN/ACK from WLC spoofing IP address 17.253.5.207
TCP ACK packet to 17.253.53.207
HTTP GET captive.apple.com
HTTP 200 OK + Redirect URL to Web Server
DNS Query: What ip can I use for the Web Server hostname
DNS Response: Web Server hostname is at x.x.x.x
TCP SYN packet to Web Server IP address
TCP SYN/ACK from Web Server
TCP ACK packet to Web Server
HTTP(S) GET for web redirection URL
HTTP 200 OK with web authentication portal (HTTP(S))
HTTP POST Request with authentication parameters (ex: username/password)
Check user locally or against RADIUS
RADIUS Access Request
RADIUS Access Accept/Reject
Delete (Reject) client or go to RUN State (Accept)
HTTP 200 OK with authentication success (Accept)
This URL for connectivity check varies depending on the client OS
WLC Intercepts TCP connection and redirects client
If the Redirect URL contains web server hostname, the client must resolve it via DNS
Here, if the WebServer URL is HPPTS, then a TLS/SSL session is established between the client and WebServer. Client must trust WebServer certificate, otherwise you will get Certificate warning.

設定

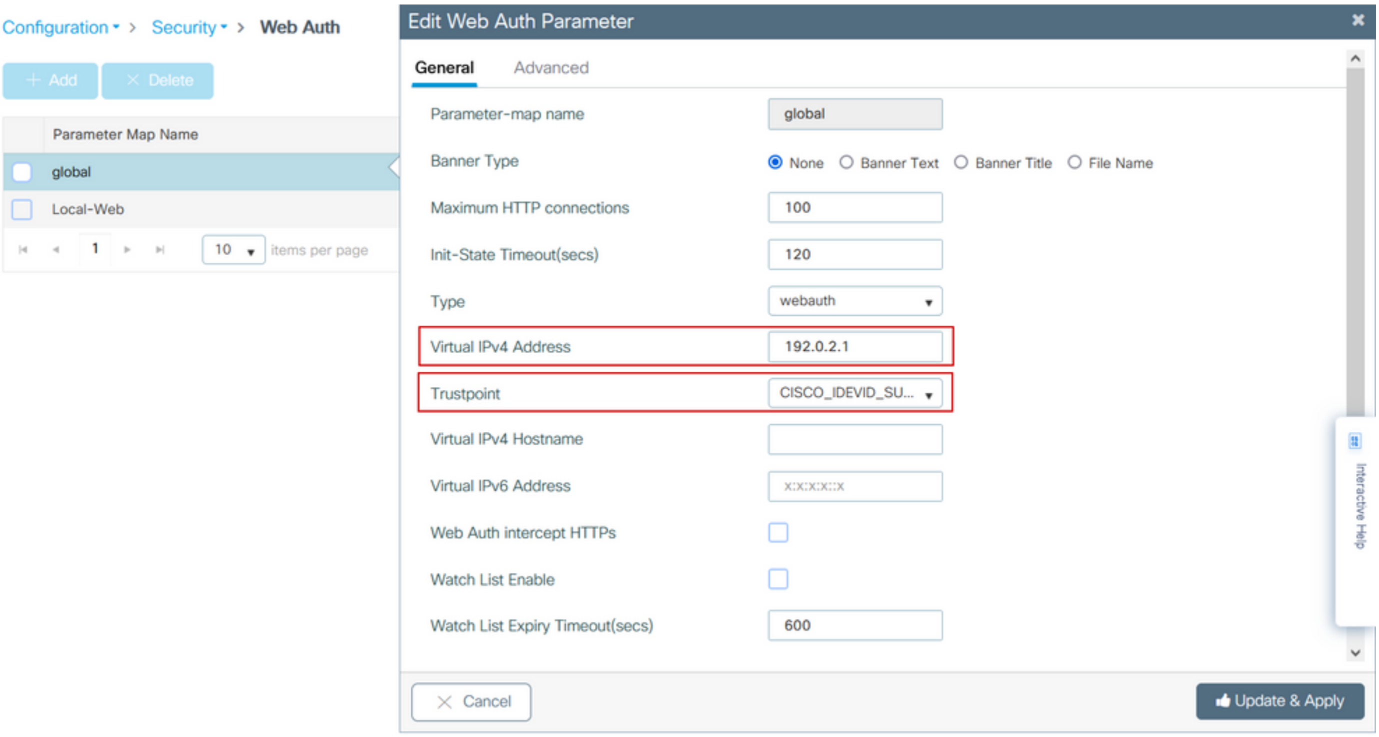
網路圖表



設定Web引數設定

步驟 1.導航到配置>安全> Web身份驗證，然後選擇全局引數對映。驗證是否已配置虛擬IPv4地址和信任點以提供正確的重定向功能。

注意：預設情況下，瀏覽器使用HTTP網站啟動重定向進程，如果需要HTTPS重定向，則必須選中Web Auth intercept HTTPs；但是，不建議使用此配置，因為它會增加CPU使用率。



Parameter-map name: global

Banner Type: ● None ○ Banner Text ○ Banner Title ○ File Name

Maximum HTTP connections: 100

Init-State Timeout(secs): 120

Type: webauth

Virtual IPv4 Address: 192.0.2.1

Trustpoint: CISCO_IDEVID_SU...

Virtual IPv4 Hostname:

Virtual IPv6 Address: X:X:X:X::X

Web Auth intercept HTTPs: □

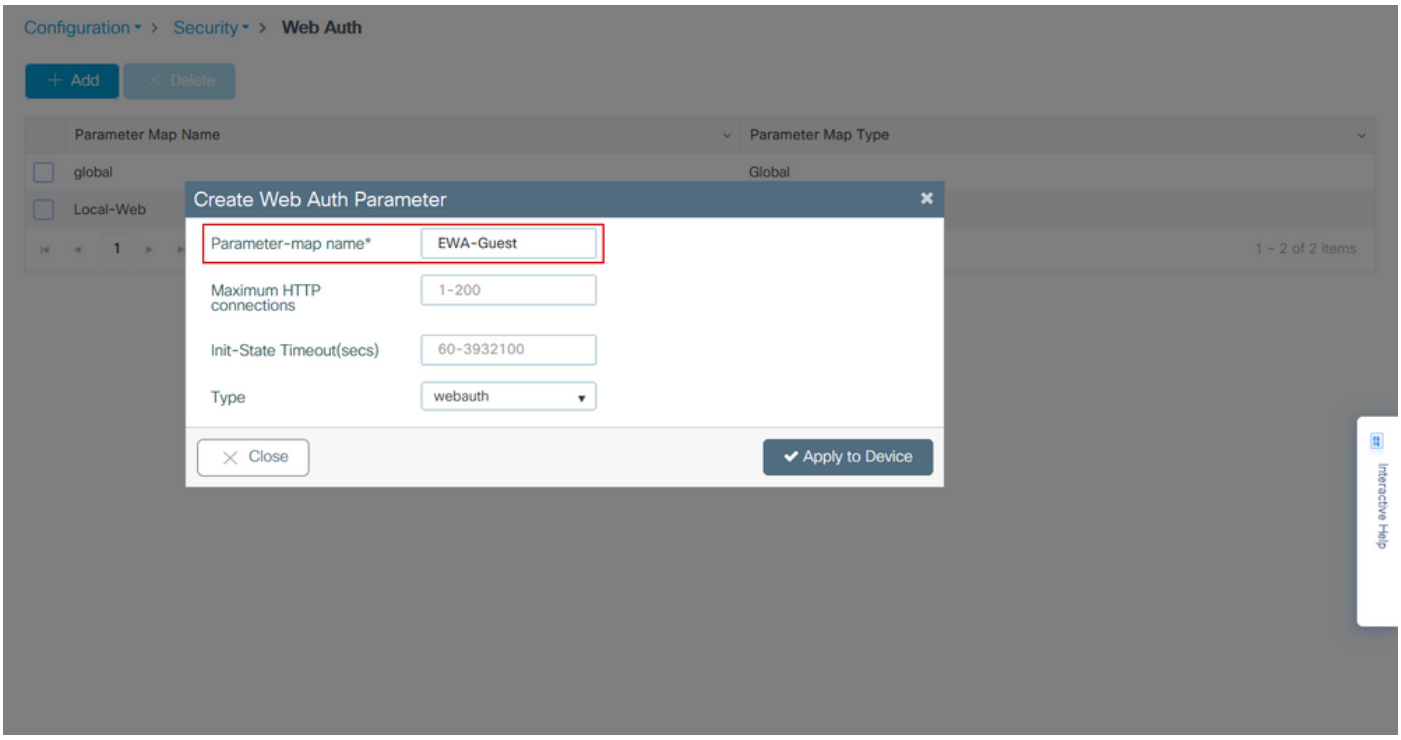
Watch List Enable: □

Watch List Expiry Timeout(secs): 600

CLI配置：

```
<#root>
9800#
configure terminal
9800(config)#
parameter-map type webauth global
9800(config-params-parameter-map)#
virtual-ip ipv4 192.0.2.1
9800(config-params-parameter-map)#
trustpoint CISCO_IDEVID_SUDI

9800(config-params-parameter-map)#
secure-webauth-disable

9800(config-params-parameter-map)#
webauth-http-enable
```

步驟 2.選擇+ Add 並配置指向外部伺服器的新引數對映的名稱。或者，配置排除客戶端之前的HTTP身份驗證失敗的最大次數以及客戶端可以保持Web身份驗證狀態的時間（以秒為單位）。


Configuration > Security > Web Auth

Create Web Auth Parameter

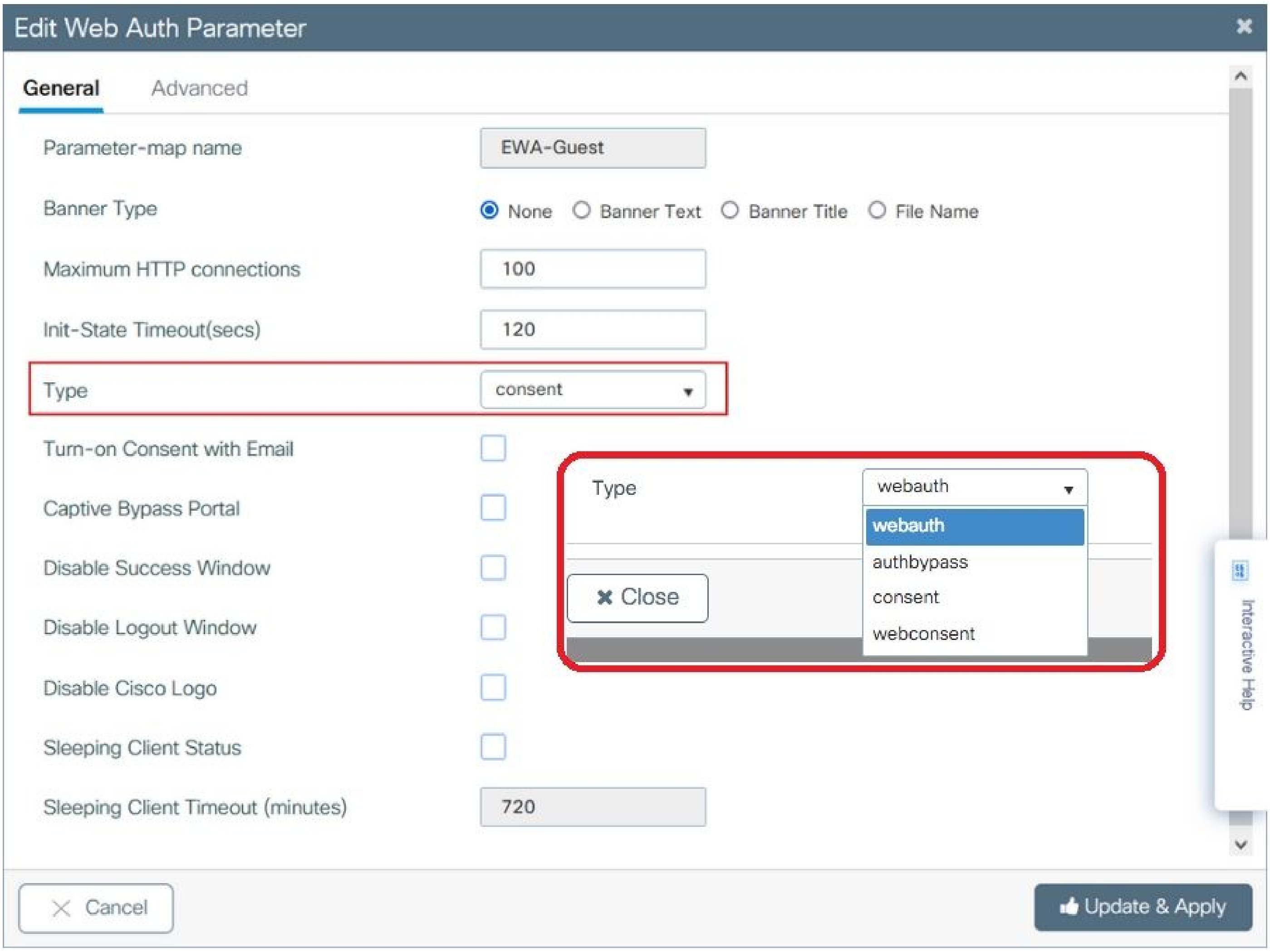
Parameter-map name*: EWA-Guest

Maximum HTTP connections: 1-200

Init-State Timeout(secs): 60-3932100

Type: webauth

步驟 3.選擇新建立的引數對映，在General 頁籤中從Type下拉選單配置身份驗證型別。


Edit Web Auth Parameter

General | Advanced

Parameter-map name: EWA-Guest

Banner Type: ● None ○ Banner Text ○ Banner Title ○ File Name

Maximum HTTP connections: 100

Init-State Timeout(secs): 120

Type: consent

Turn-on Consent with Email: □

Captive Bypass Portal: □

Disable Success Window: □

Disable Logout Window: □

Disable Cisco Logo: □

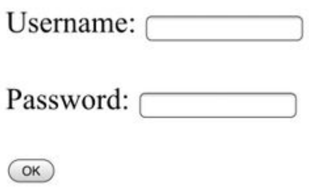
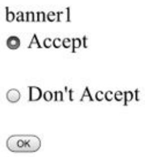
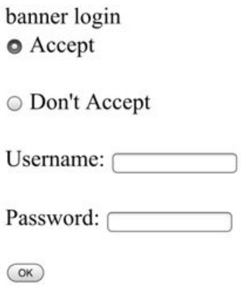
Sleeping Client Status: □

Sleeping Client Timeout (minutes): 720

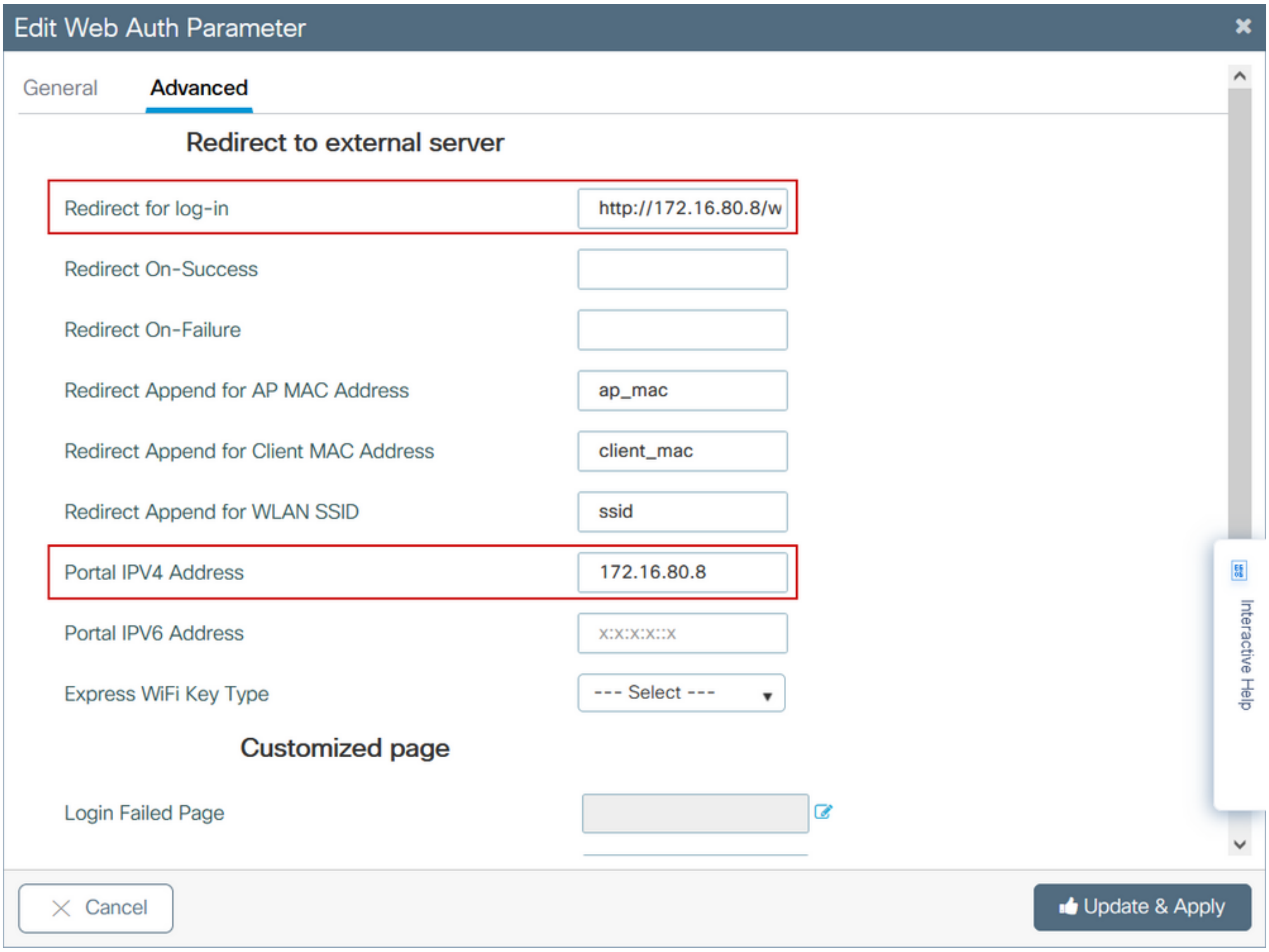
Type: webauth

- Parameter-map name =分配給WebAuth引數對映的名稱

- Maximum HTTP connections =排除客戶端之前身份驗證失敗的次數
- Init-State Timeout (secs) =客戶端可以處於Web身份驗證狀態的秒
- 型別= Web身份驗證的型別

webauth	authbypass	同意	網路同意
 Username: ▭ Password: ▭ OK	客戶端連線到 SSID並取得IP位址，然後取得9800 WLC 檢查MAC地址 允許輸入 網路，如果是，則將其移動 到RUN狀態（如果不是） 不允許加入。 （不會回到Web驗證）	 banner1 ● Accept ○ Don't Accept OK	 banner login ● Accept ○ Don't Accept Username: ▭ Password: ▭ OK

步驟 4. 在Advanced頁籤中，分別使用特定伺服器站點URL和IP地址配置登入和門戶IPV4地址的重定向。



Edit Web Auth Parameter

General | Advanced

Redirect to external server

Redirect for log-in: http://172.16.80.8/w

Redirect On-Success:

Redirect On-Failure:

Redirect Append for AP MAC Address: ap_mac

Redirect Append for Client MAC Address: client_mac

Redirect Append for WLAN SSID: ssid

Portal IPV4 Address: 172.16.80.8

Portal IPV6 Address: X:X:X:X::X

Express WiFi Key Type: --- Select ---

Customized page

Login Failed Page

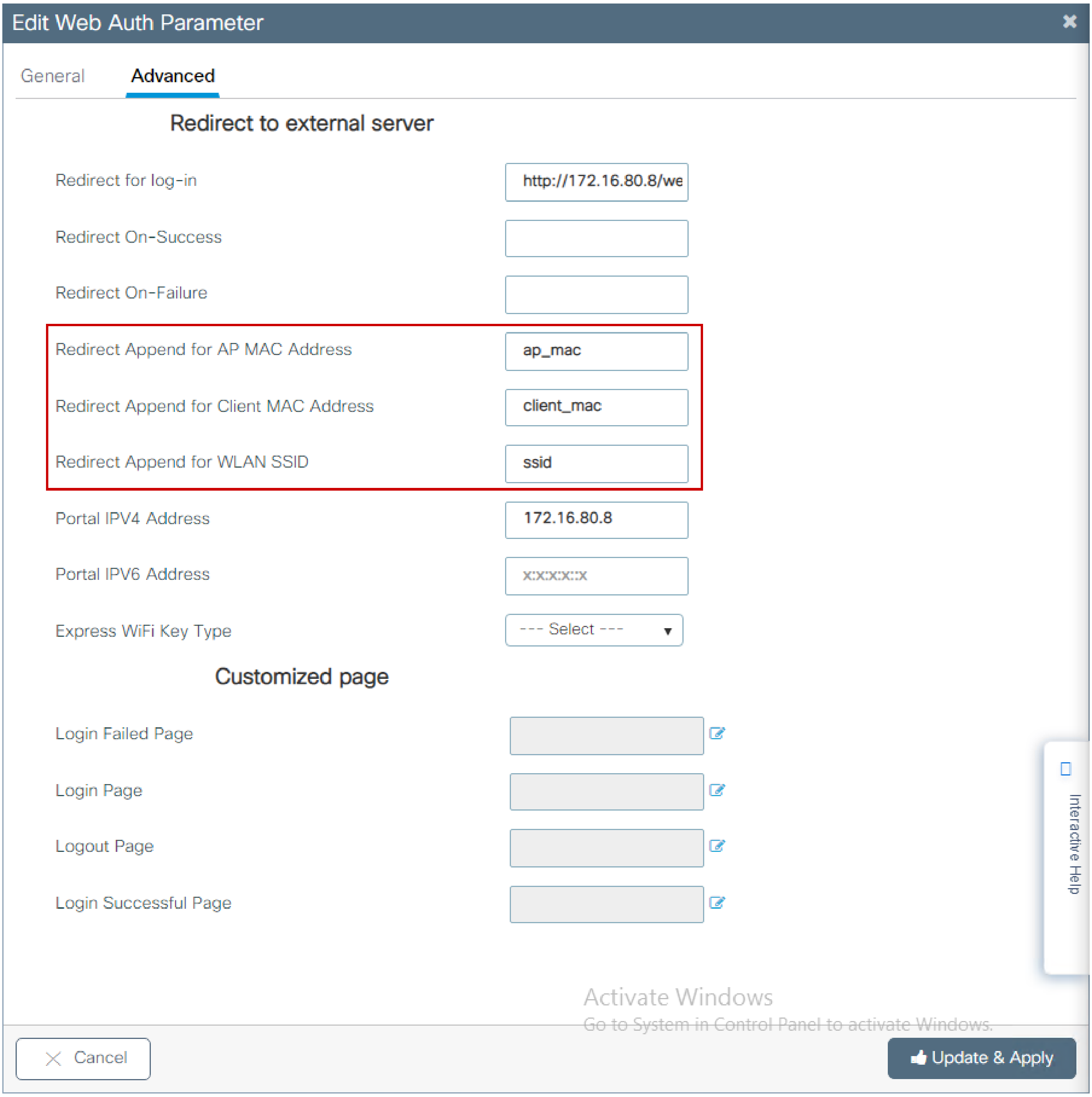
步驟2、3和4的CLI配置：

<#root>

```
9800(config)#
parameter-map type webauth EWA-Guest
9800(config-params-parameter-map)#
type consent
9800(config-params-parameter-map)#
redirect for-login http://172.16.80.8/webauth/login.html
9800(config-params-parameter-map)#
redirect portal ipv4 172.16.80.8
```

步驟5.（可選）WLC可透過查詢字串傳送其他引數。這通常是使9800與第三方外部門戶相容的要求。欄位「Redirect Append for AP MAC Address」、「Redirect Append for Client MAC Address」和「Redirect Append for WLAN SSID」允許使用自訂名稱將其他引數附加到重新導向ACL。 選擇新建立的引數對映，導航到Advanced 頁籤，配置所需引數的名稱。可用的引數有：

- AP MAC地址（採用aa：bb：cc：dd：ee：ff格式）
- 客戶端MAC地址（採用aa：bb：cc：dd：ee：ff格式）
- SSID名稱



Redirect to external server

Redirect for log-in	http://172.16.80.8/we
Redirect On-Success	
Redirect On-Failure	
Redirect Append for AP MAC Address	ap_mac
Redirect Append for Client MAC Address	client_mac
Redirect Append for WLAN SSID	ssid
Portal IPV4 Address	172.16.80.8
Portal IPV6 Address	x:x:x:x::x
Express WiFi Key Type	--- Select ---

Customized page

Login Failed Page	
Login Page	
Logout Page	
Login Successful Page	

CLI配置：

<#root>

```
9800(config)#
```

`parameter-map type webauth EWA-Guest`

```
9800(config-params-parameter-map)#
```

```
redirect append ap-mac tag ap_mac
```

```
9800(config-params-parameter-map)#
```

```
redirect append wlan-ssid tag ssid
```

```
9800(config-params-parameter-map)#
```

```
redirect append client-mac tag client_mac
```

對於此示例，傳送到客戶端的重定向URL會導致：

```
 http://172.16.80.8/webauth/consent.html?switch_url=http://192.0.2.1/login.html&ap_mac=&ssid=&client_mac
```

注意：增加門戶IPV4地址資訊時，會自動增加一個允許從無線客戶端到外部Web身份驗證伺服器的HTTP和HTTPS流量的ACL，因此您無需配置任何額外的預先身份驗證ACL。如果您希望允許多個IP地址或URL，則唯一的選項是配置URL過濾器，以便在進行身份驗證之前允許任何與給定URL匹配的IP。除非使用URL過濾器，否則無法靜態增加多個門戶IP地址。

注意：全局引數對映是唯一一個可以定義虛擬IPv4和IPv6地址、Webauth攔截HTTP、強制繞行門戶、監視清單啟用和監視清單到期超時設定的對映。

CLI配置摘要：

本機Web伺服器

```
parameter-map type webauth <web-parameter-map-name>
 type { webauth | authbypass | consent | webconsent }
 timeout init-state sec 300
 banner text ^Cbanner login^C
```

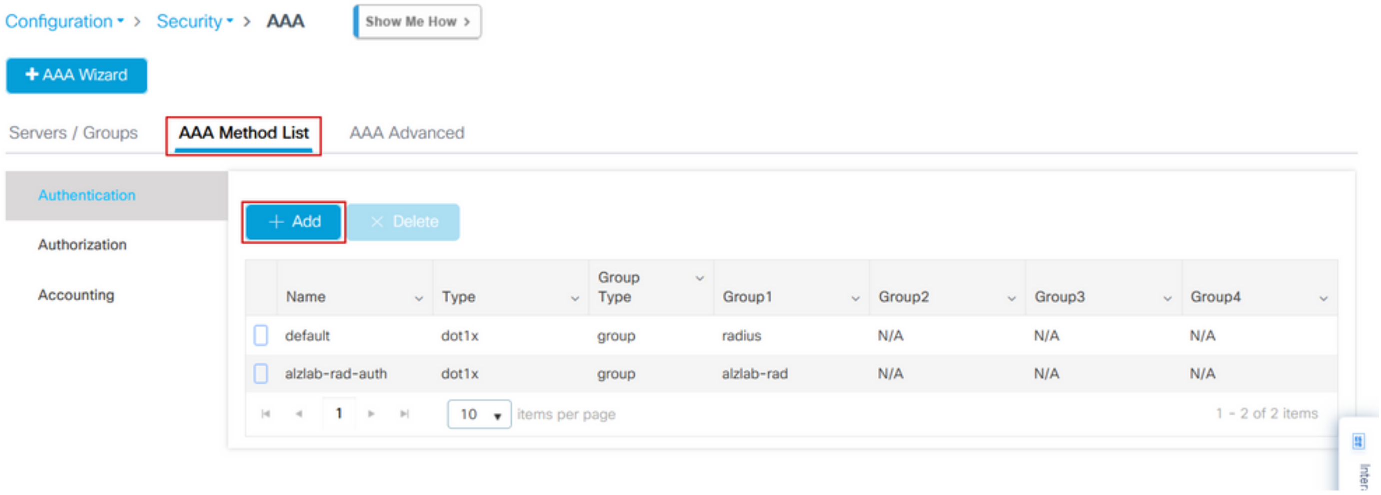
外部Web伺服器

```
parameter-map type webauth <web-parameter-map-name>
 type webauth
 timeout init-state sec 300
 redirect for-login <URL-for-webauth>
 redirect portal ipv4 <external-server's-IP
 max-http-conns 10
```

配置AAA設定

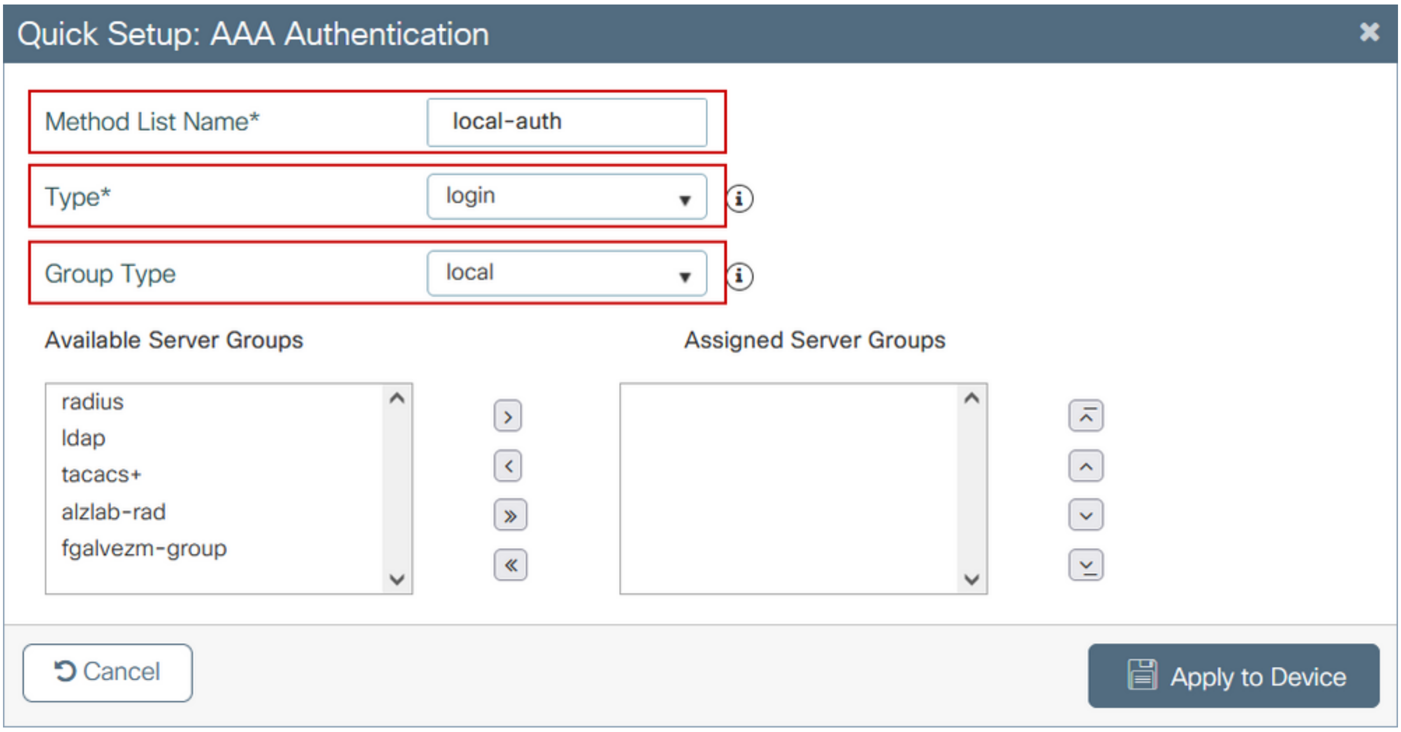
只有針對webauth或webconsent驗證型別設定的引數對應才需要此組態區段。

步驟 1.導航到Configuration > Security > AAA，然後選擇AAA Method List。配置新的方法清單，選擇+增加並填寫清單詳細資訊；確保「Type」設定為「login」，如下圖所示。



	Name	Type	Group Type	Group1	Group2	Group3	Group4
□	default	dot1x	group	radius	N/A	N/A	N/A
□	alzlab-rad-auth	dot1x	group	alzlab-rad	N/A	N/A	N/A

1 · 10 items per page · 1 - 2 of 2 items



Method List Name*: local-auth

Type*: login

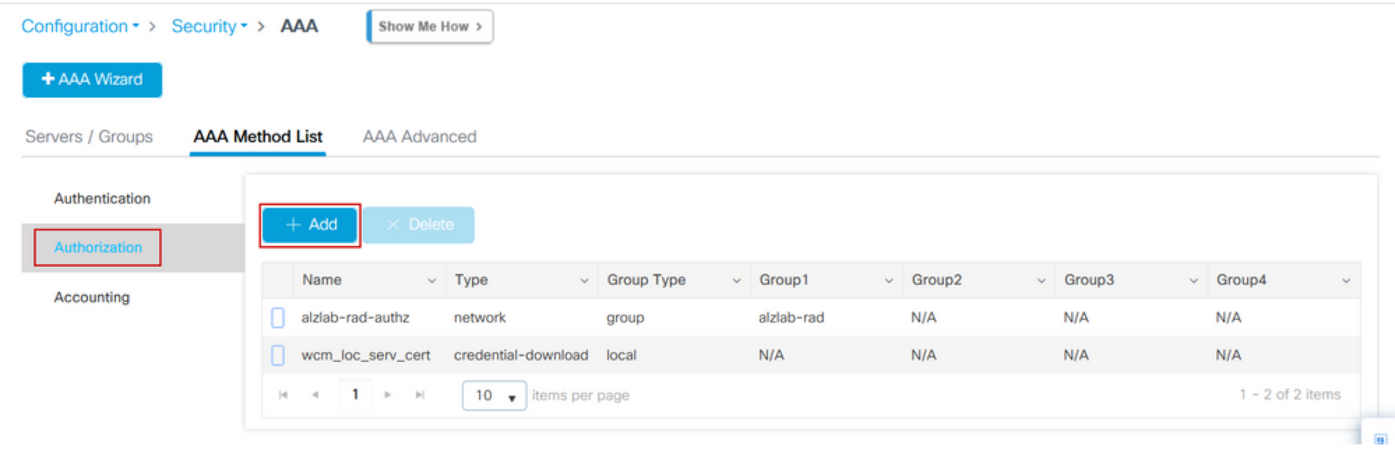
Group Type: local

Available Server Groups:
- radius
- ldap
- tacacs+
- alzlab-rad
- fgalvezm-group

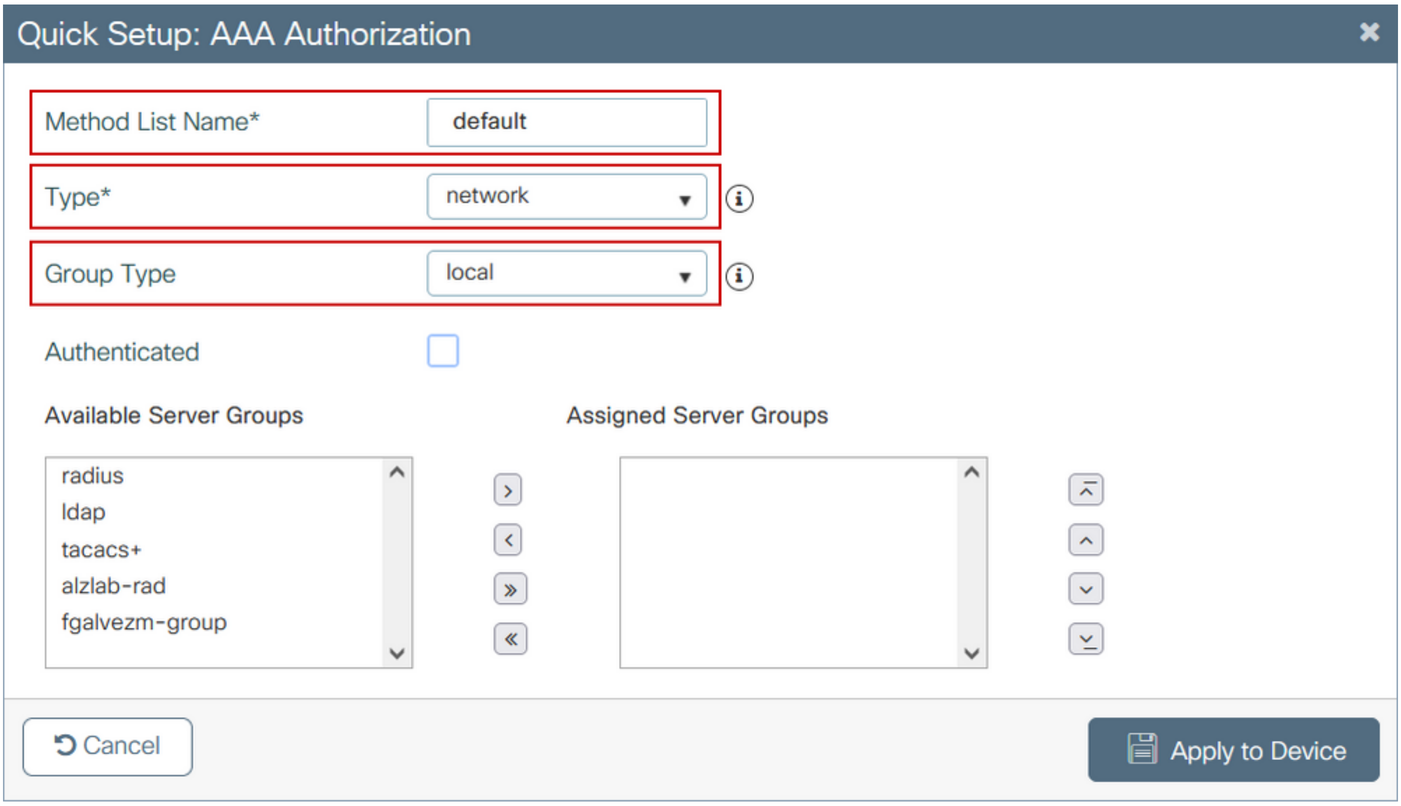
Assigned Server Groups:

步驟 2.選擇授權，然後選擇+增加以建立新方法清單。將其命名為default，並將Type命名為network，如圖所示。

注意：由於WLAN第3層安全配置：為了使本地登入方法清單正常工作，請確保裝置中存在配置「aaa authorization network default local」。這意味著必須定義名為default的授權方法清單才能正確配置本地Web身份驗證。本部分中配置了此特定的授權方法清單。



Name	Type	Group Type	Group1	Group2	Group3	Group4
alzlab-rad-authz	network	group	alzlab-rad	N/A	N/A	N/A
wcm_loc_serv_cert	credential-download	local	N/A	N/A	N/A	N/A

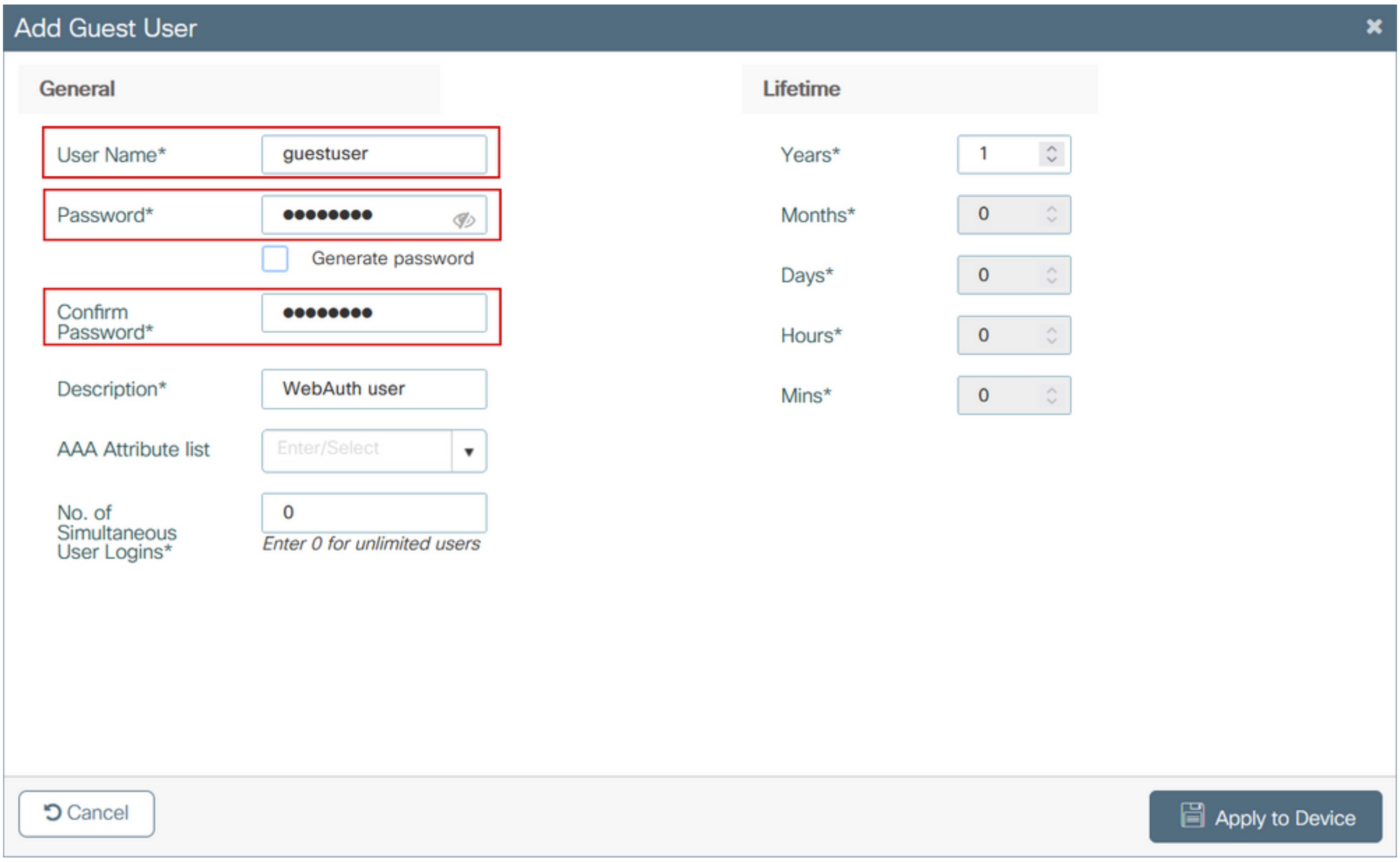


步驟1和2的CLI配置：

```
<#root>
9800(config)#

aaa new-model

9800(config)#

aaa authentication login local-auth local

9800(config)#

aaa authorization network default local
```

注意：如果需要外部RADIUS身份驗證，請閱讀以下與9800 WLC上的RADIUS伺服器配置相關的說明：9800 WLC上的AAA配置。確保身份驗證方法清單將「login」設定為type而不是dot1x。

步驟 3.導航到配置>安全>訪客使用者。選擇+ Add並配置訪客使用者帳戶詳細資訊。



General

User Name*	guestuser
Password*	●●●●●●●●
□	Generate password
Confirm Password*	●●●●●●●●
Description*	WebAuth user
AAA Attribute list	Enter/Select ▾
No. of Simultaneous User Logins*	0 *Enter 0 for unlimited users*

Lifetime

Years*	1
Months*	0
Days*	0
Hours*	0
Mins*	0

CLI配置：

<#root>

```
9800(config)#

user-name guestuser


9800(config-user-name)#

description "WebAuth user"


9800(config-user-name)#

password 0 <password>


9800(config-user-name)#

type network-user description "WebAuth user" guest-user lifetime year 1


If permanent users are needed then use this command:
9800(config)#

username guestuserperm privilege 0 secret 0 <password>
```

步驟4.（可選）根據引數對應定義，會自動建立兩個存取控制清單(ACL)。這些ACL是用來定義哪些流量會觸發重新導向至Web伺服器，以及允許哪些流量通過。如果存在特定要求（例如多個

Web伺服器IP地址或URL過濾器），請導航到Configuration > Security > ACL，選擇+ Add並定義必要的規則；當deny語句定義流量透過時，會重定向允許語句。

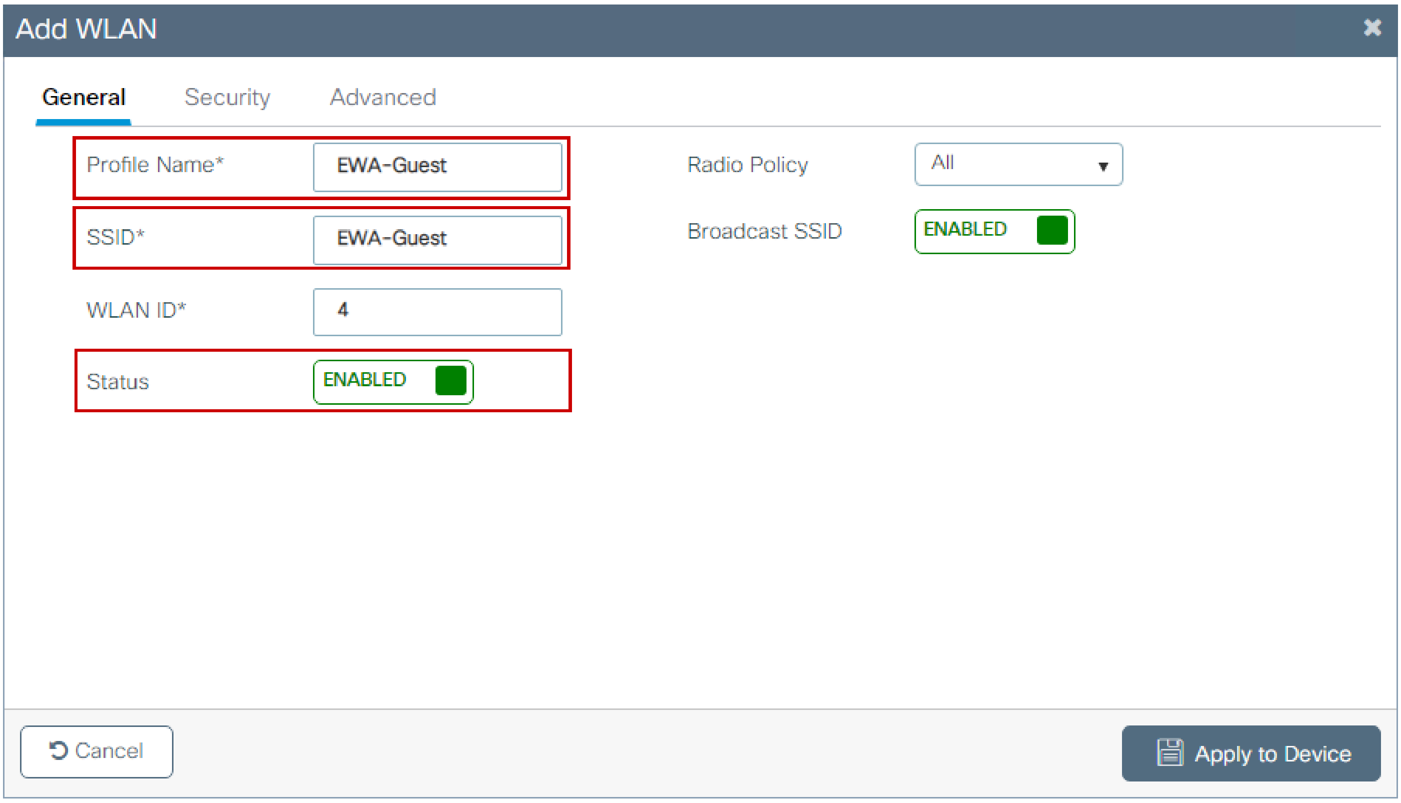
自動建立的ACL規則包括：

<#root>

```
alz-9800#

show ip access-list


Extended IP access list WA-sec-172.16.80.8
10 permit tcp any host 172.16.80.8 eq www
20 permit tcp any host 172.16.80.8 eq 443
30 permit tcp host 172.16.80.8 eq www any
40 permit tcp host 172.16.80.8 eq 443 any
50 permit tcp any any eq domain
60 permit udp any any eq domain
70 permit udp any any eq bootpc
80 permit udp any any eq bootps
90 deny ip any any (1288 matches)
Extended IP access list WA-v4-int-172.16.80.8
10 deny tcp any host 172.16.80.8 eq www
20 deny tcp any host 172.16.80.8 eq 443
30 permit tcp any any eq www
40 permit tcp any host 192.0.2.1 eq 443
```

配置策略和標籤

步驟 1.導航到配置>標籤和配置檔案> WLANs，選擇+增加以建立新的WLAN。在常規頁籤中定義配置檔案和SSID名稱以及狀態。



Profile Name* EWA-Guest

SSID* EWA-Guest

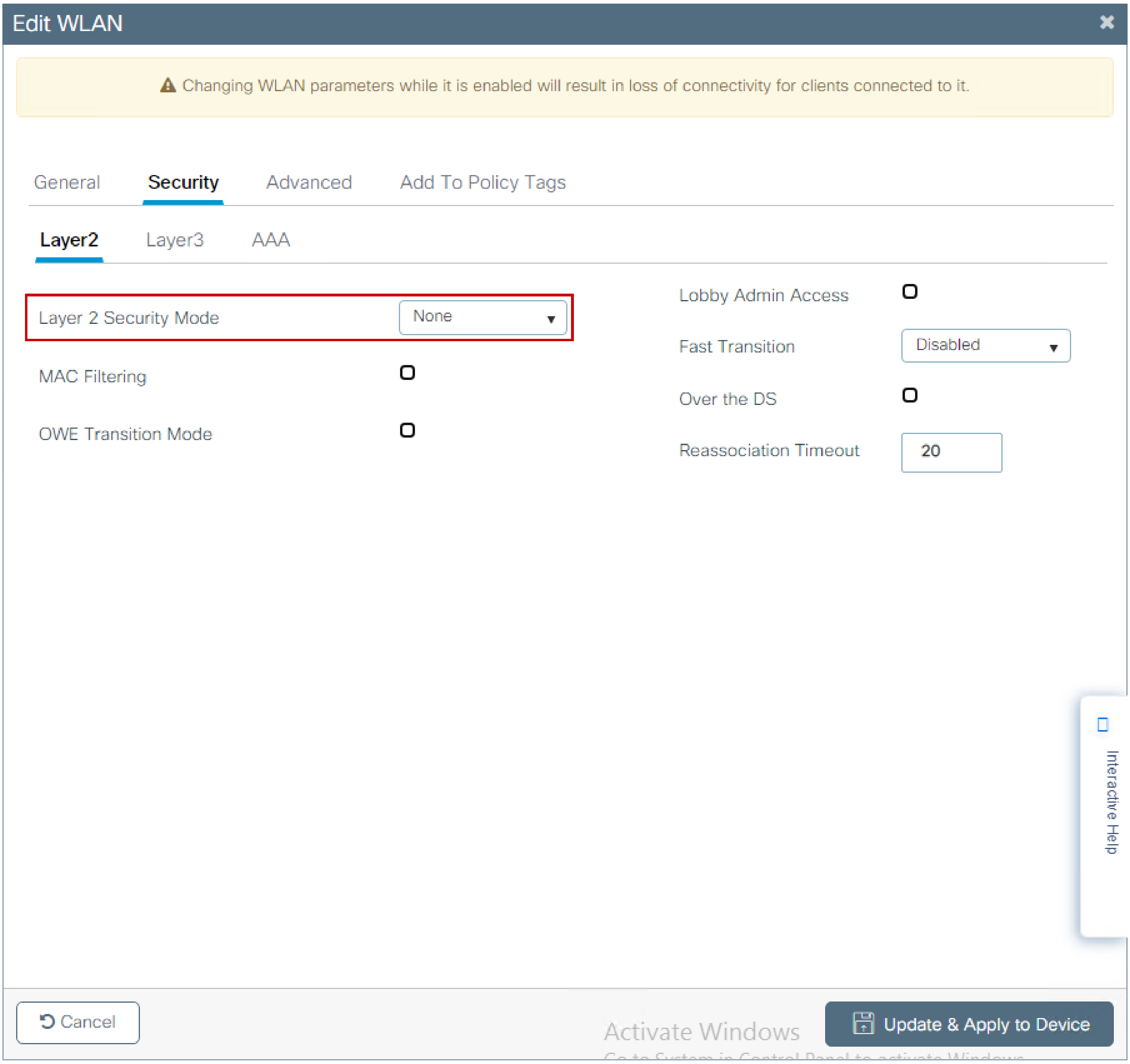
WLAN ID* 4

Status ENABLED

Radio Policy All

Broadcast SSID ENABLED

步驟 2.如果不需要任何無線加密機制，請選擇Security頁籤，並將第2層身份驗證設定為None。在Layer 3頁籤中，選中Web Policy框，從下拉選單中選擇引數對映，然後從下拉選單中選擇身份驗證清單。或者，如果之前定義了自定義ACL，請選擇Show Advanced Settings，然後從下拉選單中選擇適當的ACL。



Edit WLAN

⚠ Changing WLAN parameters while it is enabled will result in loss of connectivity for clients connected to it.

Layer 2 Security Mode: None

MAC Filtering □

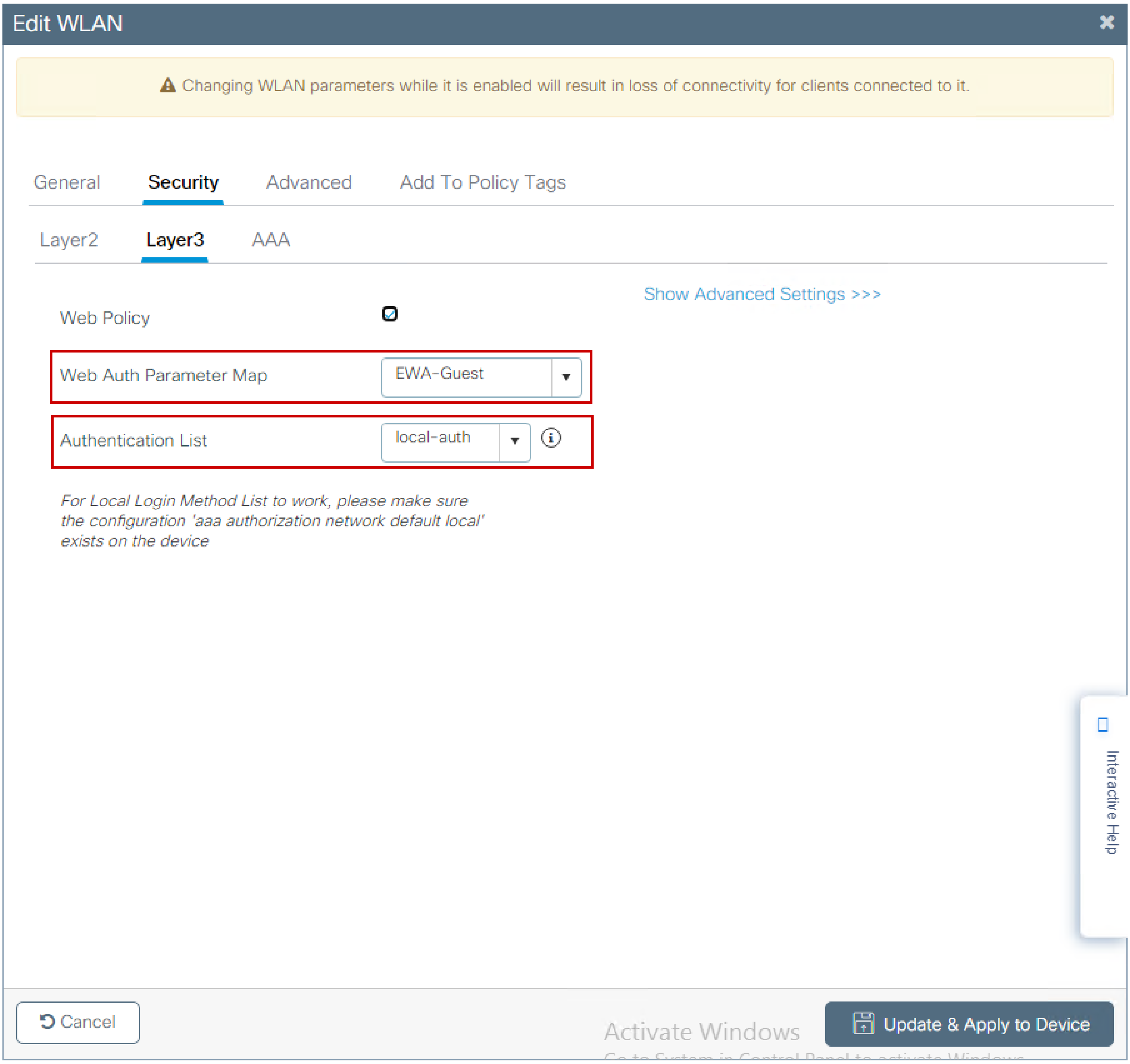
OWE Transition Mode □

Lobby Admin Access □

Fast Transition: Disabled

Over the DS □

Reassociation Timeout: 20


Edit WLAN
⚠ Changing WLAN parameters while it is enabled will result in loss of connectivity for clients connected to it.
General
Security
Advanced
Add To Policy Tags
Layer2
Layer3
AAA
Show Advanced Settings >>>
Interactive Help
Cancel
Update & Apply to Device
Activate Windows

Web Policy ☑

Web Auth Parameter Map: EWA-Guest

Authentication List: local-auth

For Local Login Method List to work, please make sure the configuration 'aaa authorization network default local' exists on the device

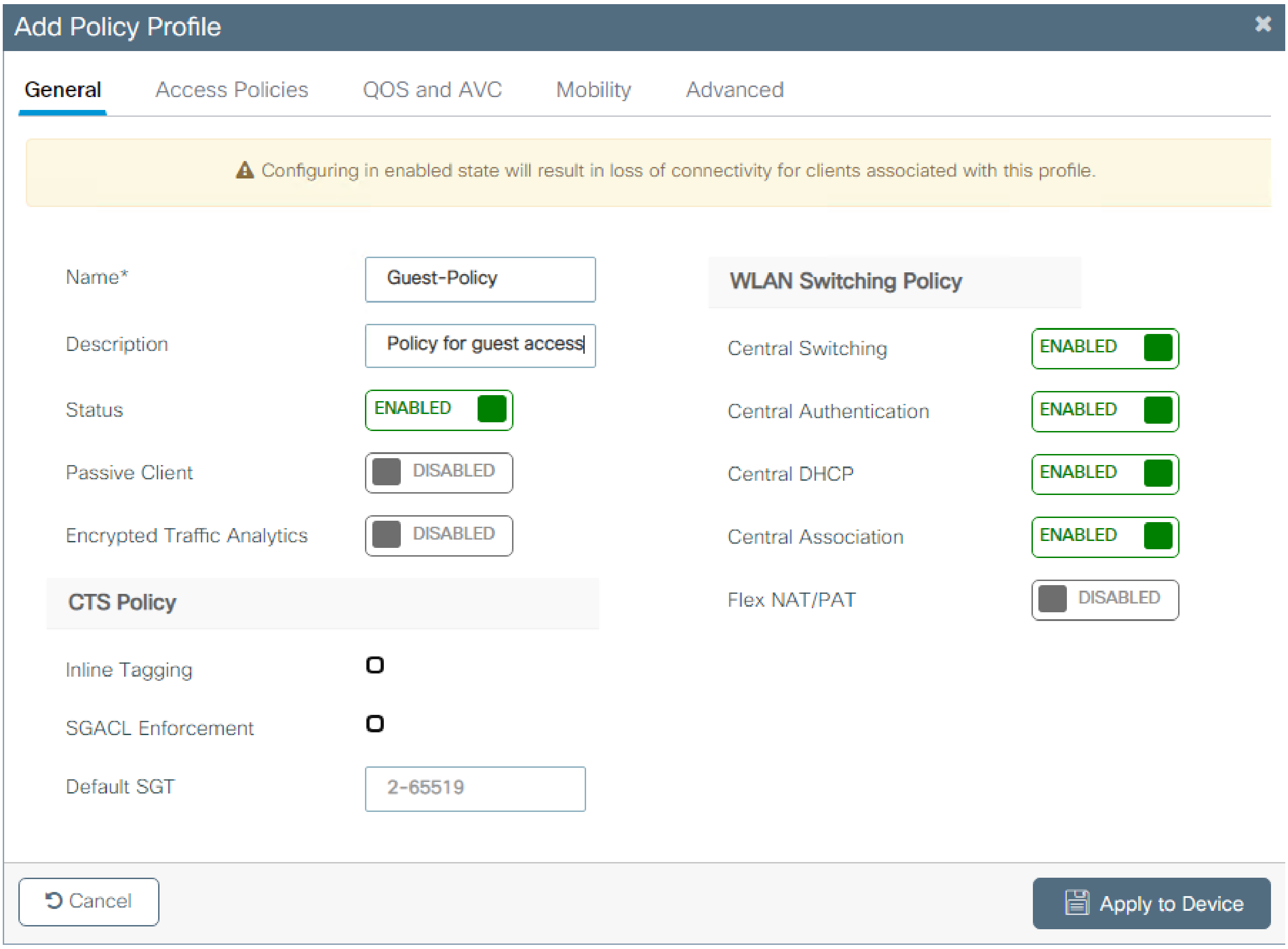
CLI配置：

<#root>

```
9800(config)#

wlan EWA-Guest 4 EWA-Guest

9800(config-wlan)#

no security ft adaptive

9800(config-wlan)#

no security wpa

9800(config-wlan)#
```

```
no security wpa wpa2

9800(config-wlan)#

no security wpa wpa2 ciphers aes

9800(config-wlan)#

no security wpa akm dot1x

9800(config-wlan)#

security web-auth

9800(config-wlan)#

security web-auth authentication-list local-auth

9800(config-wlan)#

security web-auth parameter-map EWA-Guest

9800(config-wlan)#

 no shutdown
```

步驟 3.導航到配置>標籤和配置檔案>策略，選擇+增加。定義策略名稱和狀態；確保為本地模式AP啟用了WLAN交換策略下的集中設定。在Access Policies 頁籤中，從VLAN/VLAN Group下拉選單中選擇正確的VLAN，如圖所示。



⚠ Configuring in enabled state will result in loss of connectivity for clients associated with this profile.

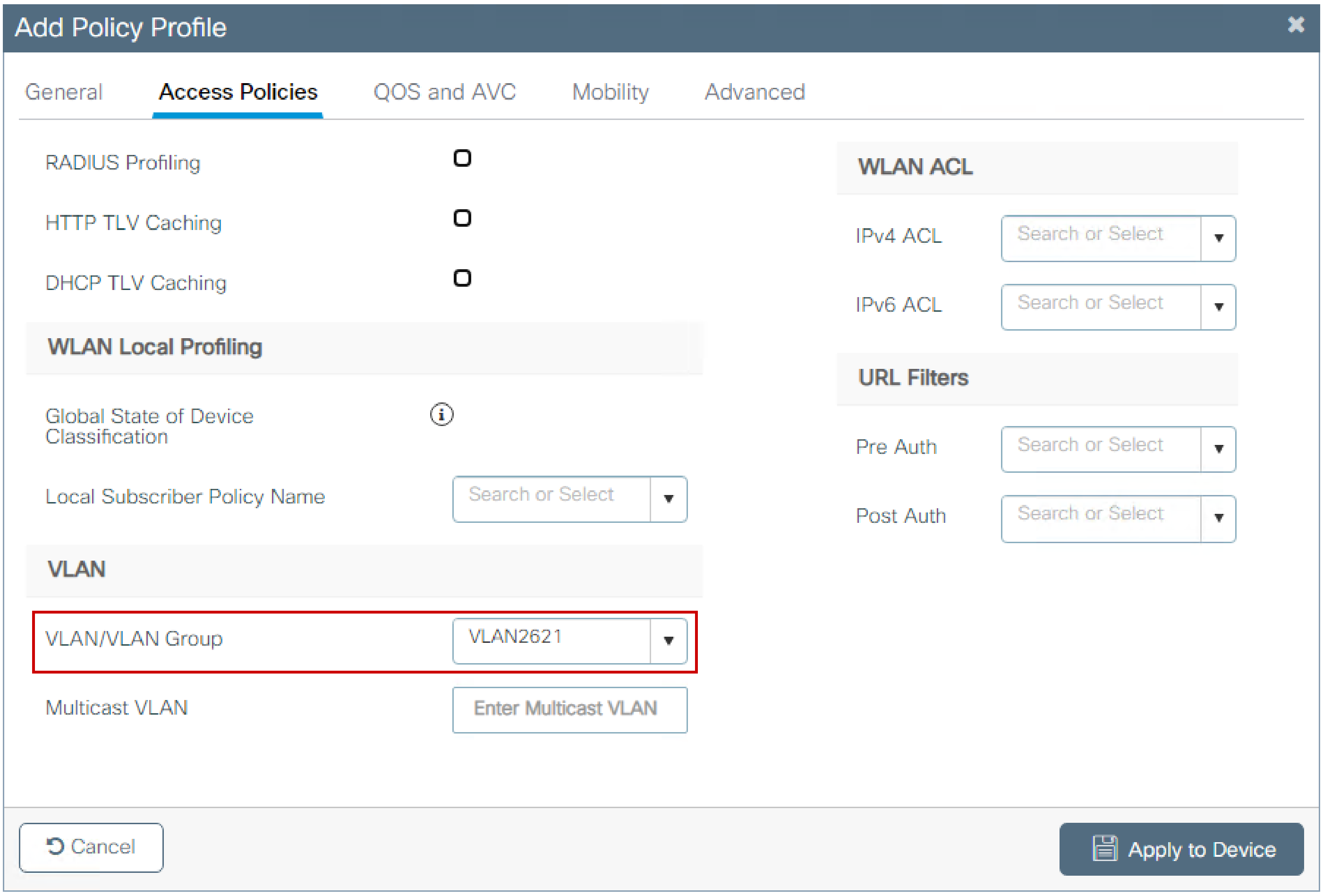
Name*	Guest-Policy
Description	Policy for guest access
Status	ENABLED
Passive Client	DISABLED
Encrypted Traffic Analytics	DISABLED

CTS Policy

Inline Tagging	□
SGACL Enforcement	□
Default SGT	2-65519

WLAN Switching Policy

Central Switching	ENABLED
Central Authentication	ENABLED
Central DHCP	ENABLED
Central Association	ENABLED
Flex NAT/PAT	DISABLED



Add Policy Profile

RADIUS Profiling ☐

HTTP TLV Caching ☐

DHCP TLV Caching ☐

WLAN Local Profiling

Global State of Device Classification

Local Subscriber Policy Name: Search or Select

VLAN

VLAN/VLAN Group: VLAN2621

Multicast VLAN: Enter Multicast VLAN

WLAN ACL

IPv4 ACL: Search or Select

IPv6 ACL: Search or Select

URL Filters

Pre Auth: Search or Select

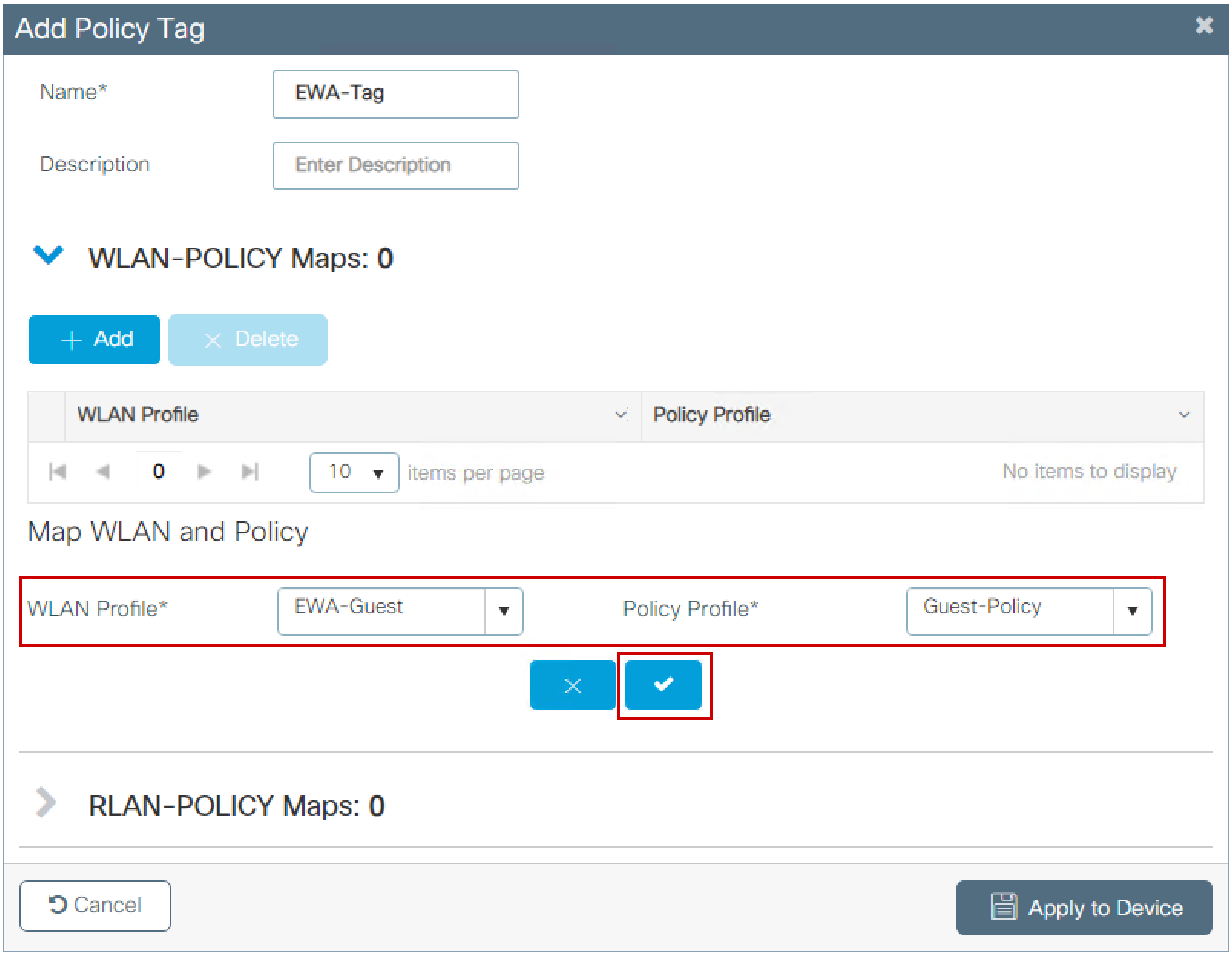
Post Auth: Search or Select

CLI配置：

<#root>

```
9800(config)#

wireless profile policy Guest-Policy


9800(config-wireless-policy)#

description "Policy for guest access"


9800(config-wireless-policy)#

vlan VLAN2621


9800(config-wireless-policy)#

no shutdown
```

步驟 4.導航到配置>標籤和配置檔案>標籤，在策略頁籤中依次選擇+增加。定義標籤名稱，然後在WLAN-POLICY Maps下選擇+ Add，並增加之前建立的WLAN和策略配置檔案。

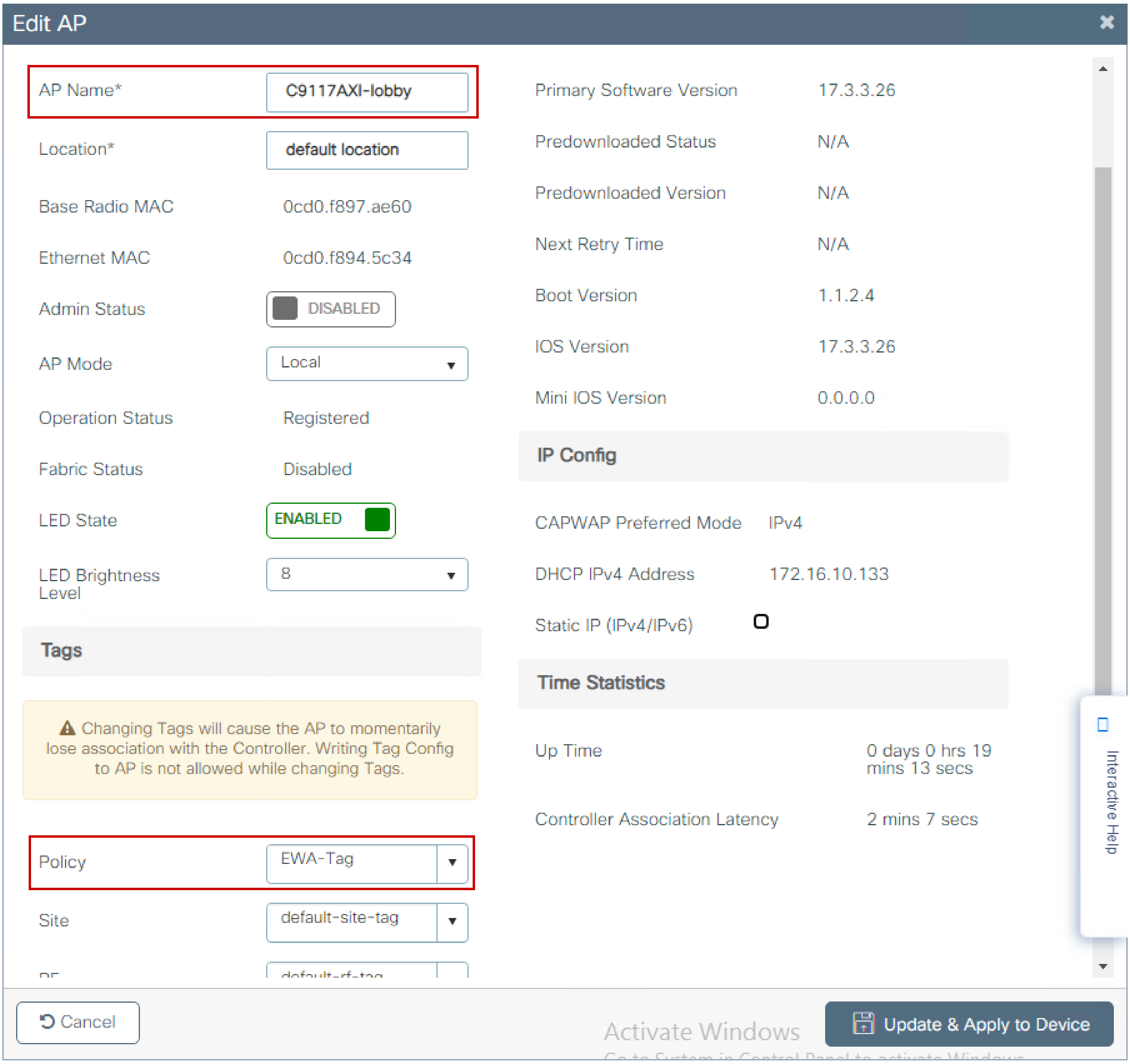


CLI配置：

<#root>

```
9800(config)#

wireless tag policy EWA-Tag


9800(config-policy-tag)#

wlan EWA-Guest policy Guest-Policy
```

步驟 5.導航到Configuration > Wireless > Access Points，並選擇用於廣播此SSID的AP。從Edit AP選單中，從「Policy」下拉選單中選擇新建立的標籤。



Edit AP

AP Name* C9117AXI-lobby

Location* default location

Base Radio MAC 0cd0.f897.ae60

Ethernet MAC 0cd0.f894.5c34

Admin Status DISABLED

AP Mode Local

Operation Status Registered

Fabric Status Disabled

LED State ENABLED

LED Brightness Level 8

Tags

⚠ Changing Tags will cause the AP to momentarily lose association with the Controller. Writing Tag Config to AP is not allowed while changing Tags.

Policy EWA-Tag

Site default-site-tag

RF default-rf-tag

Primary Software Version 17.3.3.26

Predownloaded Status N/A

Predownloaded Version N/A

Next Retry Time N/A

Boot Version 1.1.2.4

IOS Version 17.3.3.26

Mini IOS Version 0.0.0.0

IP Config

CAPWAP Preferred Mode IPv4

DHCP IPv4 Address 172.16.10.133

Static IP (IPv4/IPv6) □

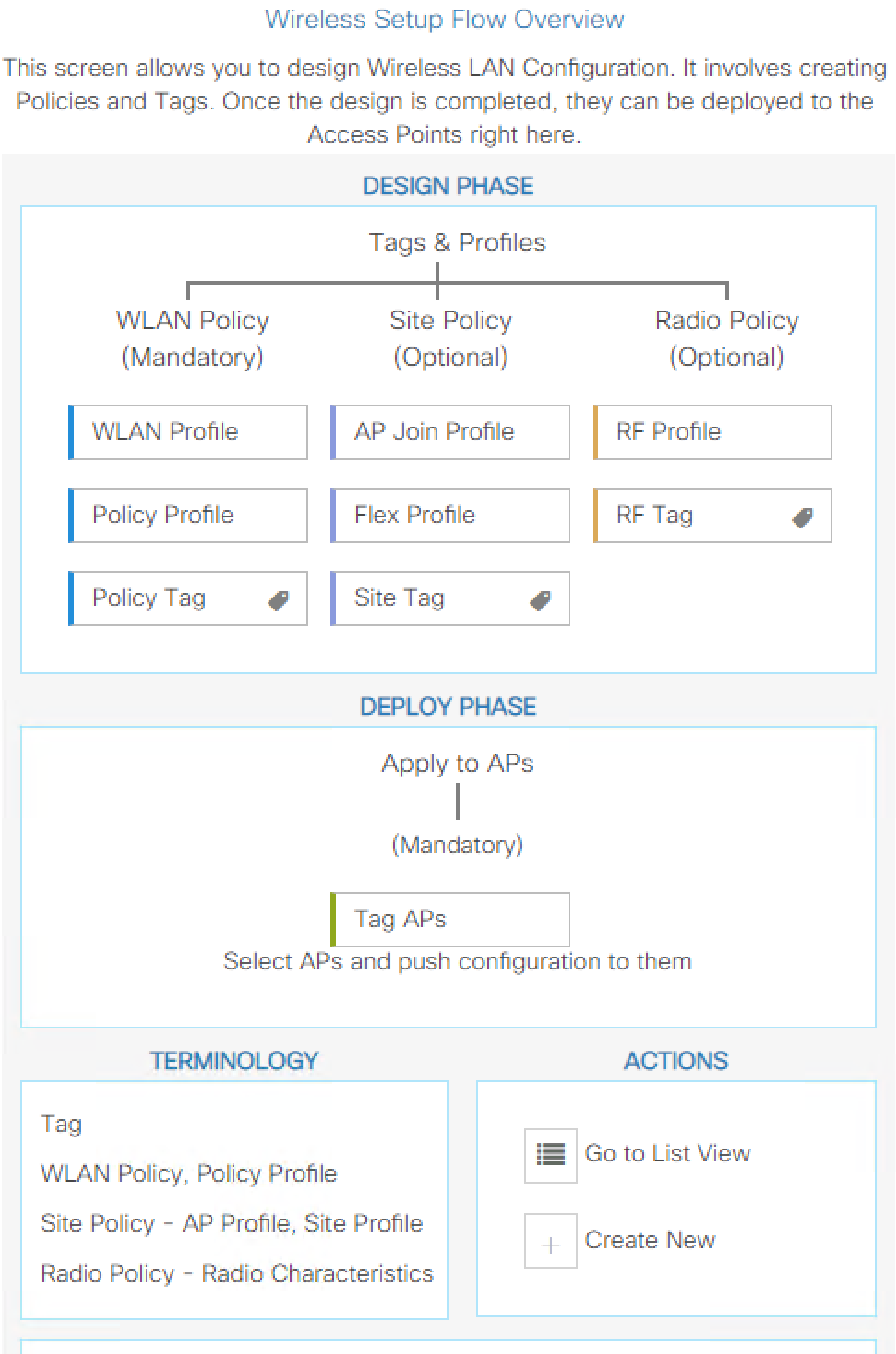
Time Statistics

Up Time 0 days 0 hrs 19 mins 13 secs

Controller Association Latency 2 mins 7 secs

如果需要同時標籤多個AP，則有兩個可用選項：

選項A。導航到配置>無線設定>高級，從此處選擇立即開始以顯示配置選單清單。選擇標籤AP旁邊的清單圖示，這會顯示處於加入狀態的所有AP的清單，檢查所需的AP，然後選擇+標籤AP，從下拉選單中選擇建立的策略標籤。



Wireless Setup Flow Overview

This screen allows you to design Wireless LAN Configuration. It involves creating Policies and Tags. Once the design is completed, they can be deployed to the Access Points right here.

DESIGN PHASE

DEPLOY PHASE

Select APs and push configuration to them

TERMINOLOGY

Tag

WLAN Policy, Policy Profile

Site Policy - AP Profile, Site Profile

Radio Policy - Radio Characteristics

ACTIONS

```
----------------------------------------------------------------------------------------------------------------------------
0x7B 0x73 0x0B 0x1E 0x46 0x2A 0xD7 0x8F 0x23 0xF3 0xFE 0x9E 0x5C 0xB0 0xEB 0xF8 0x0000000a


0x0000001a 1



9800#

show platform software cgacl chassis active F0 group-idx <group index> acl


Acl ID Acl Name CGACL Type Protocol Direction Sequence
----------------------------------------------------------------------------------------------------------------------------
16 IP-Adm-V6-Int-ACL-global Punt IPv6 IN 1

25 WA-sec-172.16.80.8 Security IPv4 IN 2



26 WA-v4-int-172.16.80.8 Punt IPv4 IN 1


19 implicit_deny Security IPv4 IN 3
21 implicit_deny_v6 Security IPv6 IN 3
18 preauth_v6 Security IPv6 IN 2
```

疑難排解

永遠線上跟蹤

WLC 9800提供永遠開啟追蹤功能。這可確保所有與客戶端連線相關的錯誤、警告和通知級別消息被持續記錄，並且您可以在事件發生後檢視事件或故障條件的日誌。

註：根據生成的日誌量，您可以將時間從幾小時縮短到幾天。

為了檢視9800 WLC預設收集的跟蹤，您可以透過SSH/Telnet連線到9800 WLC並閱讀以下步驟（確保將會話記錄到文本檔案中）。

步驟 1.檢查控制器目前時間，以便追蹤問題發生時的記錄。

```
<#root>

9800#

show clock
```

步驟 2.根據系統配置的指示，從控制器緩衝區或外部系統日誌收集系統日誌。這樣可以快速檢視系統運行狀況和錯誤（如果有）。

```
<#root>

9800#

show logging
```

步驟 3.驗證是否啟用了任何調試條件。

```
<#root>

9800#

show debugging

IOSXE Conditional Debug Configs:
Conditional Debug Global State: Stop
IOSXE Packet Tracing Configs:
Packet Infra debugs:
Ip Address                                               Port
------------------------------------------------------|----------
```

注意：如果發現列出任何條件，則意味著所有遇到啟用條件（MAC地址、IP地址等）的進程的跟蹤將記錄到調試級別。如此可能會增加記錄量。因此，建議在不主動調試時清除所有條件。

步驟 4.假設步驟3中沒有列出測試中的mac地址作為條件。收集特定MAC位址的永遠線上通知層級追蹤。

```
<#root>

9800#

show logging profile wireless filter [mac | ip] [<aaaa.bbbb.cccc> | <a.b.c.d>] to-file always-on-<FILENA
```

您可顯示作業階段中的內容，或可將檔案複製到外部 TFTP 伺服器。

```
<#root>

9800#

 more bootflash:always-on-<FILENAME.txt>


or
9800#

copy bootflash:always-on-<FILENAME.txt> tftp://<a.b.c.d>/<path>/always-on-<FILENAME.txt>
```

條件式偵錯和無線電主動式追蹤

如果永遠線上的追蹤無法提供足夠資訊來判斷觸發調查中問題的原因，您可以啟用條件式偵錯並擷取「無線電作用中(RA)」追蹤，此追蹤會為與指定條件（此案例為使用者端mac位址）互動的所有處理作業提供偵錯層級追蹤。要啟用條件調試，請閱讀以下步驟。

步驟 1.確保沒有啟用調試條件。

```
<#root>

9800#

clear platform condition all
```

步驟 2.為要監控的無線客戶端MAC地址啟用調試條件。

以下命令會開始監控提供的 MAC 位址 30 分鐘（1800 秒）。您可選擇將此時間增加至 2085978494 秒。

```
<#root>

9800#

debug wireless mac <aaaa.bbbb.cccc> {monitor-time <seconds>}
```

注意：要同時監控多個客戶端，請對每個mac地址運行debug wireless mac命令。

注意：無線客戶端活動不會顯示在終端會話中，因為所有日誌都緩衝在內部以便以後檢視。

步驟 3.重現您要監控的問題或行為。

步驟 4.如果在預設或配置的監控時間之前重現問題，則停止調試。

```
<#root>

9800#

no debug wireless mac <aaaa.bbbb.cccc>
```

當監控時間結束或偵錯無線停止後，9800 WLC 會產生本機檔案，名稱如下：

ra_trace_MAC_aaaabbbbcccc_HHMMSS.XXX_timezone_DayWeek_Month_Day_year.log

步驟 5. 收集 MAC 位址活動的檔案。 您可將 ra_trace.log 複製到外部伺服器，或將輸出內容直接

顯示於螢幕上。

檢查 RA 追蹤檔案的名稱。

```
<#root>

9800#

dir bootflash: | inc ra_trace
```

將檔案複製到外部伺服器：

```
<#root>

9800#

copy bootflash:ra_trace_MAC_aaaabbbbcccc_HHMMSS.XXX_timezone_DayWeek_Month_Day_year.log tftp://<a.b.c.d>
```

顯示內容：

```
<#root>

9800#

more bootflash:ra_trace_MAC_aaaabbbbcccc_HHMMSS.XXX_timezone_DayWeek_Month_Day_year.log
```

步驟 6.如果根本原因仍不明顯，請收集內部日誌，這些日誌是調試級別日誌的更詳細檢視。您不需要再次調試客戶端，因為命令提供已收集並在內部儲存的調試日誌。

```
<#root>

9800#

show logging profile wireless internal filter [mac | ip] [<aaaa.bbbb.cccc> | <a.b.c.d>] to-file ra-inter
```

> 注意：此命令輸出返回所有進程的所有日誌記錄級別的跟蹤，而且輸出量非常大。請與Cisco TAC聯絡以幫助分析這些跟蹤。

```
<#root>

9800#

copy bootflash:ra-internal-<FILENAME>.txt tftp://<a.b.c.d>/ra-internal-<FILENAME>.txt
```

顯示內容：

```
<#root>

9800#

more bootflash:ra-internal-<FILENAME>.txt
```

步驟 7.移除偵錯條件。

注意：請確保在排除會話故障後始終刪除調試條件。

內嵌封包擷取

9800控制器可以本機偵測封包；這可以更輕鬆地排解疑難問題，因為控制平面封包處理可視性。

步驟 1.定義ACL以過濾相關的流量。對於Web身份驗證，建議允許來自和傳向Web伺服器的流量，以及來自和傳向客戶端連線的一些AP的流量。

```
<#root>

9800(config)#

ip access-list extended EWA-pcap

9800(config-ext-nacl)#

permit ip any host <web server IP>

9800(config-ext-nacl)#

permit ip host <web server IP> any

9800(config-ext-nacl)#

permit ip any host <AP IP>

9800(config-ext-nacl)#

permit ip host <AP IP> any
```

步驟 2.定義監視捕獲引數。確保兩個方向的控制平面流量均已啟用，介面是指控制器的物理上行鏈路。

```
<#root>

9800#
```

```
monitor capture EWA buffer size <buffer size in MB>


9800#

monitor capture EWA access-list EWA-pcap


9800#

monitor capture EWA control-plane both interface <uplink interface> both
```

<#root>

```
9800#

show monitor capture EWA


Status Information for Capture EWA
Target Type:
Interface: Control Plane, Direction: BOTH

Interface: TenGigabitEthernet0/1/0, Direction: BOTH


Status : Inactive
Filter Details:

Access-list: EWA-pcap


Inner Filter Details:
Buffer Details:
Buffer Type: LINEAR (default)

Buffer Size (in MB): 100


Limit Details:
Number of Packets to capture: 0 (no limit)
Packet Capture duration: 0 (no limit)
Packet Size to capture: 0 (no limit)
Packet sampling rate: 0 (no sampling)
```

步驟 3.開始監控器捕獲並重現問題。

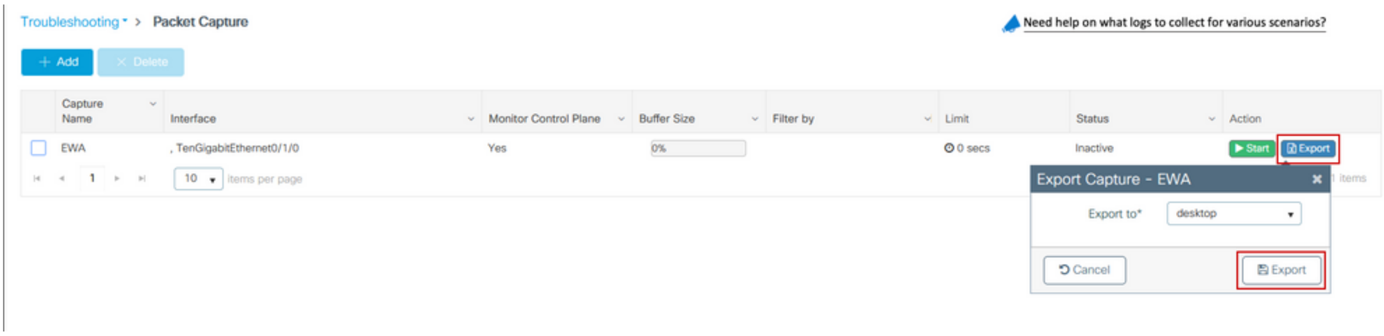
<#root>

```
9800#

monitor capture EWA start


Started capture point : EWA
```

步驟 4.停止監控器捕獲並導出它。

```
<#root>

9800#

monitor capture EWA stop


Stopped capture point : EWA
9800#monitor capture EWA export tftp://<a.b.c.d>/EWA.pcap
```

或者，也可以從GUI下載捕獲，導航到故障排除>資料包捕獲，並在配置的捕獲上選擇導出。從下拉選單中選擇desktop（案頭），透過HTTP將捕獲下載到所需的資料夾中。


Troubleshooting > Packet Capture
Need help on what logs to collect for various scenarios?
+ Add
× Delete
Start
Export
1
10 items per page
1 items
Export Capture - EWA
Cancel
Export

	Capture Name	Interface	Monitor Control Plane	Buffer Size	Filter by	Limit	Status	Action
□	EWA	, TenGigabitEthernet0/1/0	Yes	0%		0 secs	Inactive	

Export to* desktop

客戶端故障排除

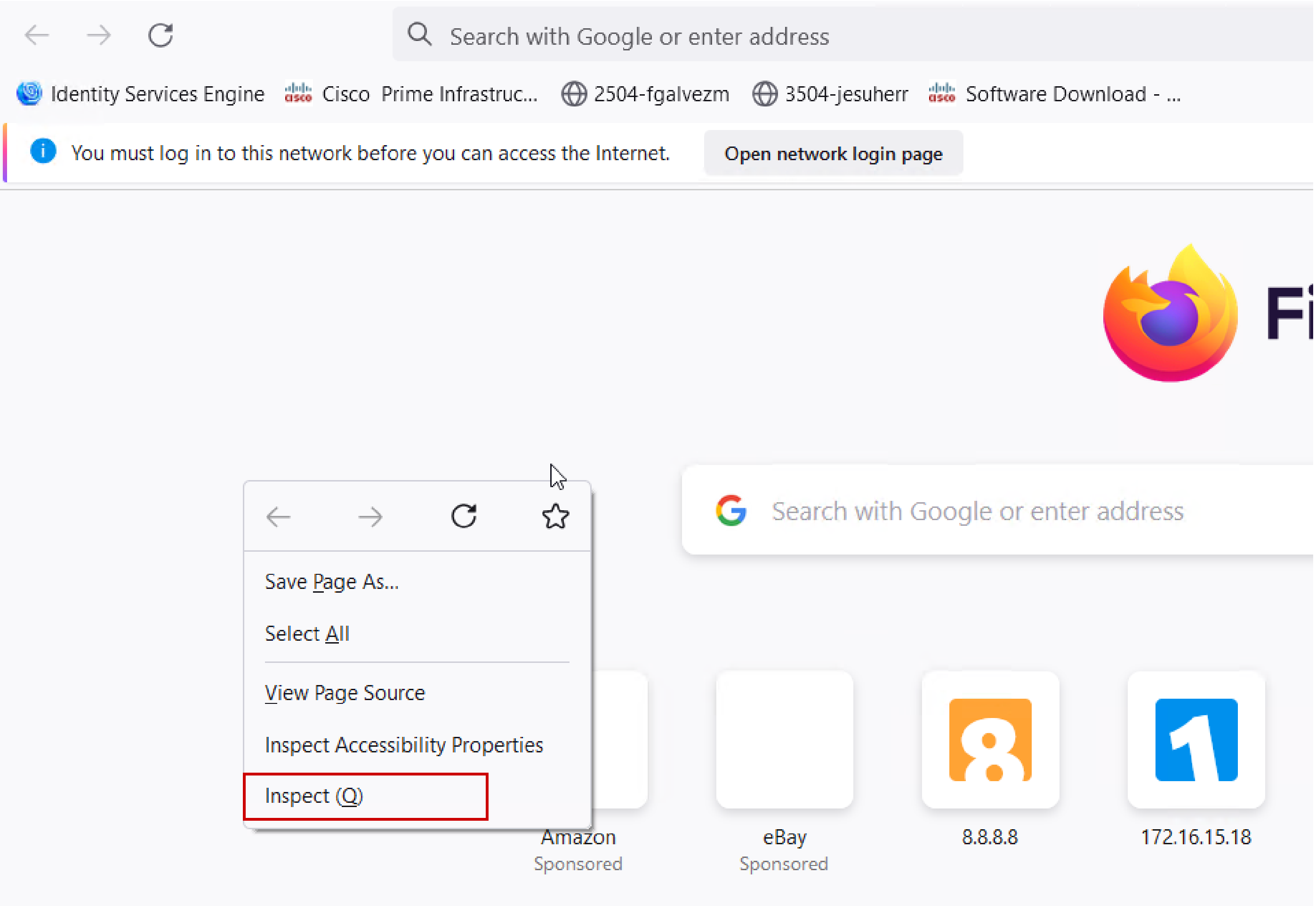
Web驗證WLAN依賴於客戶端行為，在此基礎上，客戶端行為知識和資訊是確定Web驗證錯誤行為的根本原因的關鍵。

HAR瀏覽器故障排除

許多現代的瀏覽器，例如Mozilla Firefox和Google Chrome，提供控制檯開發人員工具來調試Web應用程式互動。HAR檔案是客戶端-伺服器互動的記錄，提供HTTP互動的時間表以及請求和響應資訊（報頭、狀態代碼、引數等）。

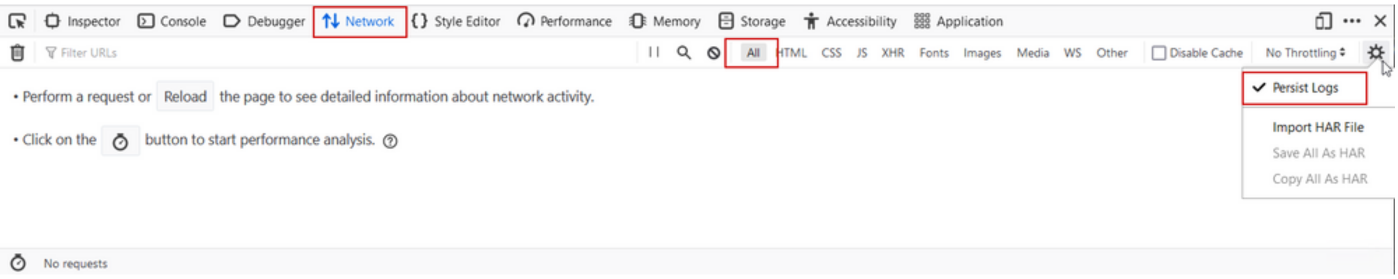
HAR檔案可以從使用者端瀏覽器匯出，並匯入到其他瀏覽器以進行進一步分析。本文檔概述了如何從Mozilla Firefox收集HAR檔案。

步驟 1.使用Ctrl + Shift + I打開Web Developer Tools，或者按一下右鍵瀏覽器內容並選擇Inspect。



You must log in to this network before you can access the Internet.

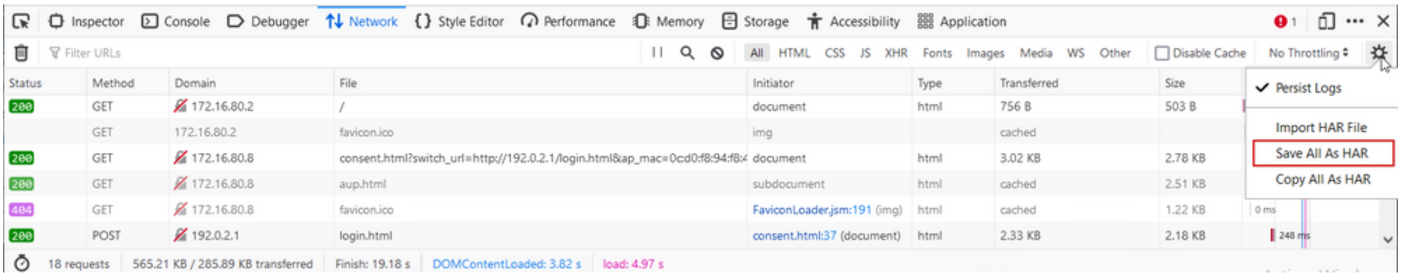
步驟 2.導航到網路，確保選擇「全部」以捕獲所有請求型別。選擇齒輪圖示並確保Persist Logs旁邊有一個箭頭，否則每當觸發域更改時，日誌請求都會被清除。



- Perform a request or Reload the page to see detailed information about network activity.
- Click on the button to start performance analysis.

No requests

步驟 3.重現問題，確保瀏覽器記錄所有請求。一旦重現問題「停止網路日誌記錄」，然後選擇齒輪圖示並選擇Save All As HAR。



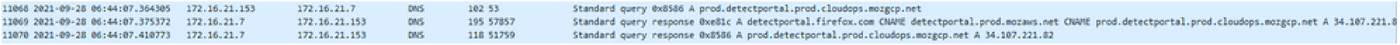
Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	172.16.80.2	/	document	html	756 B	503 B
	GET	172.16.80.2	favicon.ico	img		cached	
200	GET	172.16.80.8	consent.html?switch_url=http://192.0.2.1/login.html&ap_mac=0cd0:f8:94:f8:4	document	html	3.02 KB	2.78 KB
200	GET	172.16.80.8	aup.html	subdocument	html	cached	2.51 KB
404	GET	172.16.80.8	favicon.ico	FaviconLoader.jsm:191 (img)	html	cached	1.22 KB
200	POST	192.0.2.1	login.html	consent.html:37 (document)	html	2.33 KB	2.18 KB

18 requests | 565.21 KB / 285.89 KB transferred | Finish: 19.18 s | DOMContentLoaded: 3.82 s | load: 4.97 s

使用者端封包擷取

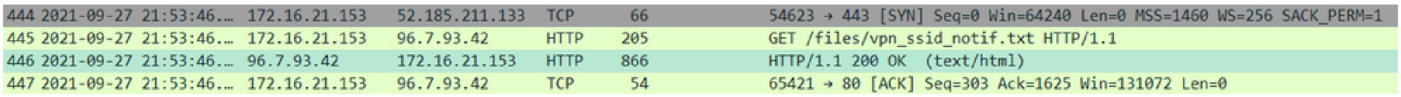
使用Windows或MacOS等作業系統的無線使用者端可以在其無線網路卡介面卡上偵測封包。雖然它們不能直接取代無線資料包捕獲，但可以大致瞭解整個Web身份驗證流程。

DNS請求：



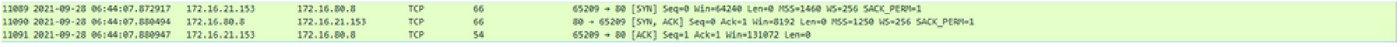
```
11068 2021-09-28 06:44:07.364305   172.16.21.153     172.16.21.7       DNS      102 53       Standard query 0x8586 A prod.detectportal.prod.cloudops.mozgcp.net
11069 2021-09-28 06:44:07.375372   172.16.21.7       172.16.21.153     DNS      195 57857    Standard query response 0xe81c A detectportal.firefox.com CNAME detectportal.prod.mozaws.net CNAME prod.detectportal.prod.cloudops.mozgcp.net A 34.107.221.8
11070 2021-09-28 06:44:07.410773   172.16.21.7       172.16.21.153     DNS      118 51759    Standard query response 0x8586 A prod.detectportal.prod.cloudops.mozgcp.net A 34.107.221.82
```

用於重定向的初始TCP握手和HTTP GET：



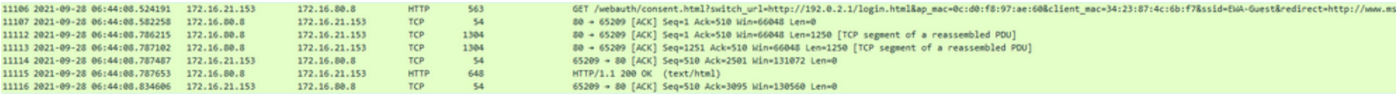
```
444 2021-09-27 21:53:46.… 172.16.21.153  52.185.211.133  TCP    66    54623 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
445 2021-09-27 21:53:46.… 172.16.21.153  96.7.93.42      HTTP   205   GET /files/vpn_ssid_notif.txt HTTP/1.1
446 2021-09-27 21:53:46.… 96.7.93.42     172.16.21.153   HTTP   866   HTTP/1.1 200 OK  (text/html)
447 2021-09-27 21:53:46.… 172.16.21.153  96.7.93.42      TCP    54    65421 → 80 [ACK] Seq=303 Ack=1625 Win=131072 Len=0
```

與外部伺服器的TCP握手：



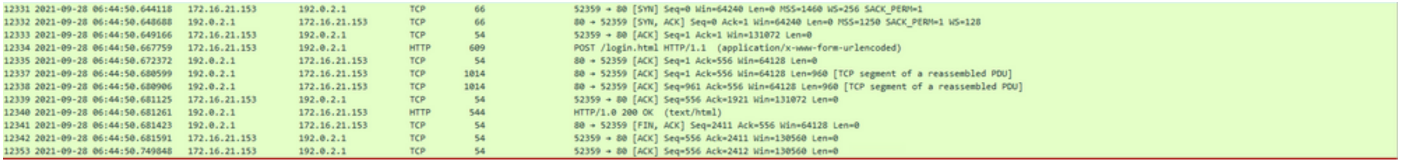
```
11089 2021-09-28 06:44:07.872917   172.16.21.153     172.16.80.8       TCP      66       65209 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
11090 2021-09-28 06:44:07.880494   172.16.80.8       172.16.21.153     TCP      66       80 → 65209 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1250 WS=256 SACK_PERM=1
11091 2021-09-28 06:44:07.880947   172.16.21.153     172.16.80.8       TCP      54       65209 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
```

HTTP GET到外部伺服器（強制網路門戶請求）：



```
11106 2021-09-28 06:44:08.524191   172.16.21.153     172.16.80.8       HTTP     563      GET /webauth/consent.html?switch_url=http://192.0.2.1/login.html&ap_mac=0c:d0:f8:97:ae:60&client_mac=34:23:87:4c:6b:f7&ssid=EWA-Guest&redirect=http://www.ms
11107 2021-09-28 06:44:08.582258   172.16.80.8       172.16.21.153     TCP       54      80 → 65209 [ACK] Seq=1 Ack=510 Win=66048 Len=0
11112 2021-09-28 06:44:08.786215   172.16.80.8       172.16.21.153     TCP     1304      80 → 65209 [ACK] Seq=1 Ack=510 Win=66048 Len=1250 [TCP segment of a reassembled PDU]
11113 2021-09-28 06:44:08.787102   172.16.80.8       172.16.21.153     TCP     1304      80 → 65209 [ACK] Seq=1251 Ack=510 Win=66048 Len=1250 [TCP segment of a reassembled PDU]
11114 2021-09-28 06:44:08.787487   172.16.21.153     172.16.80.8       TCP       54      65209 → 80 [ACK] Seq=510 Ack=2501 Win=131072 Len=0
11115 2021-09-28 06:44:08.787653   172.16.80.8       172.16.21.153     HTTP     648      HTTP/1.1 200 OK  (text/html)
11116 2021-09-28 06:44:08.834606   172.16.21.153     172.16.80.8       TCP       54      65209 → 80 [ACK] Seq=510 Ack=3095 Win=130560 Len=0
```

HTTP POST至虛擬IP以進行驗證：



```
12331 2021-09-28 06:44:50.644118   172.16.21.153     192.0.2.1         TCP       66      52359 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
12332 2021-09-28 06:44:50.648688   192.0.2.1         172.16.21.153     TCP       66      80 → 52359 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1250 SACK_PERM=1 WS=128
12333 2021-09-28 06:44:50.649166   172.16.21.153     192.0.2.1         TCP       54      52359 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
12334 2021-09-28 06:44:50.667759   172.16.21.153     192.0.2.1         HTTP     609      POST /login.html HTTP/1.1  (application/x-www-form-urlencoded)
12335 2021-09-28 06:44:50.672372   192.0.2.1         172.16.21.153     TCP       54      80 → 52359 [ACK] Seq=1 Ack=556 Win=64128 Len=0
12337 2021-09-28 06:44:50.680599   192.0.2.1         172.16.21.153     TCP     1014      80 → 52359 [ACK] Seq=1 Ack=556 Win=64128 Len=960 [TCP segment of a reassembled PDU]
12338 2021-09-28 06:44:50.680906   192.0.2.1         172.16.21.153     TCP     1014      80 → 52359 [ACK] Seq=961 Ack=556 Win=64128 Len=960 [TCP segment of a reassembled PDU]
12339 2021-09-28 06:44:50.681125   172.16.21.153     192.0.2.1         TCP       54      52359 → 80 [ACK] Seq=556 Ack=1921 Win=131072 Len=0
12340 2021-09-28 06:44:50.681261   192.0.2.1         172.16.21.153     HTTP     544      HTTP/1.0 200 OK  (text/html)
12341 2021-09-28 06:44:50.681423   192.0.2.1         172.16.21.153     TCP       54      80 → 52359 [FIN, ACK] Seq=2411 Ack=556 Win=64128 Len=0
12342 2021-09-28 06:44:50.681591   172.16.21.153     192.0.2.1         TCP       54      52359 → 80 [ACK] Seq=556 Ack=2411 Win=130560 Len=0
12353 2021-09-28 06:44:50.749848   172.16.21.153     192.0.2.1         TCP       54      52359 → 80 [ACK] Seq=556 Ack=2412 Win=130560 Len=0
```

成功嘗試的範例

這是從無線電活動跟蹤角度成功嘗試連線的輸出，使用此參考來確定連線到第3層Web身份驗證SSID的客戶端的客戶端會話階段。

802.11驗證與關聯：

<#root>

```
2021/09/28 12:59:51.781967 {wncd_x_R0-0}{1}: [client-orch-sm] [26328]: (note): MAC: 3423.874c.6bf7 Asso
2021/09/28 12:59:51.782009 {wncd_x_R0-0}{1}: [client-orch-sm] [26328]: (debug): MAC: 3423.874c.6bf7
```

Received Dot11 association request.

```
 Processing started,
```

SSID: EWA-Guest, Policy profile: Guest-Policy

```
, AP Name: C9117AXI-lobby, Ap Mac Address: 0cd0.f897.ae60 BSSID MAC0000.0000.0000 wlan ID: 4RSSI: -39,
2021/09/28 12:59:51.782152 {wncd_x_R0-0}{1}: [client-orch-state] [26328]: (note): MAC: 3423.874c.6bf7 C
2021/09/28 12:59:51.782357 {wncd_x_R0-0}{1}: [dot11-validate] [26328]: (info): MAC: 3423.874c.6bf7 WiFi
2021/09/28 12:59:51.782480 {wncd_x_R0-0}{1}: [dot11] [26328]: (debug): MAC: 3423.874c.6bf7 dot11 send a
```

Sending association response with resp_status_code: 0

```
2021/09/28 12:59:51.782483 {wncd_x_R0-0}{1}: [dot11] [26328]: (debug): MAC: 3423.874c.6bf7 Dot11 Capabi
2021/09/28 12:59:51.782509 {wncd_x_R0-0}{1}: [dot11-frame] [26328]: (info): MAC: 3423.874c.6bf7 WiFi di
2021/09/28 12:59:51.782519 {wncd_x_R0-0}{1}: [dot11] [26328]: (info): MAC: 3423.874c.6bf7 dot11 send as
2021/09/28 12:59:51.782611 {wncd_x_R0-0}{1}: [dot11] [26328]: (note): MAC: 3423.874c.6bf7
```

Association success. AID 1

```
, Roaming = False, WGB = False, 11r = False, 11w = False
2021/09/28 12:59:51.782626 {wncd_x_R0-0}{1}: [dot11] [26328]: (info): MAC: 3423.874c.6bf7 DOT11 state t
2021/09/28 12:59:51.782676 {wncd_x_R0-0}{1}: [client-orch-sm] [26328]: (debug): MAC: 3423.874c.6bf7
```

Station Dot11 association is successful.

已跳過第2層身份驗證：

<#root>

```
2021/09/28 12:59:51.782727 {wncd_x_R0-0}{1}: [client-orch-sm] [26328]: (debug): MAC: 3423.874c.6bf7 Sta
2021/09/28 12:59:51.782745 {wncd_x_R0-0}{1}: [client-orch-state] [26328]: (note): MAC: 3423.874c.6bf7 C
2021/09/28 12:59:51.782785 {wncd_x_R0-0}{1}: [client-auth] [26328]: (note): MAC: 3423.874c.6bf7
```

L2 Authentication initiated. method WEBAUTH

```
, Policy VLAN 2621,AAA override = 0
2021/09/28 12:59:51.782803 {wncd_x_R0-0}{1}: [sanet-shim-translate] [26328]: (ERR): 3423.874c.6bf7 wlan_
[...]
2021/09/28 12:59:51.787912 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7 Client
2021/09/28 12:59:51.787953 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7 Client
2021/09/28 12:59:51.787966 {wncd_x_R0-0}{1}: [client-orch-sm] [26328]: (debug): MAC: 3423.874c.6bf7
```

L2 Authentication of station is successful., L3 Authentication : 1

ACL插孔：

<#root>

```
2021/09/28 12:59:51.785227 {wncd_x_R0-0}{1}: [webauth-sm] [26328]: (info): [ 0.0.0.0]Starting Webauth,
2021/09/28 12:59:51.785307 {wncd_x_R0-0}{1}: [auth-mgr-feat_wireless] [26328]: (info): [0000.0000.0000:
2021/09/28 12:59:51.785378 {wncd_x_R0-0}{1}: [webauth-acl] [26328]: (info): capwap_9000000b[3423.874c.6
```

Applying IPv4 intercept ACL via SVM, name: WA-v4-int-172.16.80.8

```
, priority: 50, IIF-ID: 0
2021/09/28 12:59:51.785738 {wncd_x_R0-0}{1}: [epm-redirect] [26328]: (info): [0000.0000.0000:unknown]
```

URL-Redirect-ACL = WA-v4-int-172.16.80.8

```
2021/09/28 12:59:51.786324 {wncd_x_R0-0}{1}: [webauth-acl] [26328]: (info): capwap_9000000b[3423.874c.6
```

Applying IPv6 intercept ACL via SVM, name: IP-Adm-V6-Int-ACL-global, priority: 52

```
, IIF-ID: 0
2021/09/28 12:59:51.786598 {wncd_x_R0-0}{1}: [epm-redirect] [26328]: (info): [0000.0000.0000:unknown]
```

URL-Redirect-ACL = IP-Adm-V6-Int-ACL-global

```
2021/09/28 12:59:51.787904 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7 Client 
```

IP學習過程：

<#root>

```
2021/09/28 12:59:51.799515 {wncd_x_R0-0}{1}: [client-orch-state] [26328]: (note): MAC: 3423.874c.6bf7 C
2021/09/28 12:59:51.799716 {wncd_x_R0-0}{1}: [client-iplearn] [26328]: (info): MAC: 3423.874c.6bf7

IP-learn state transition: S_IPLEARN_INIT -> S_IPLEARN_IN_PROGRESS


2021/09/28 12:59:51.802213 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7 Client 
2021/09/28 12:59:51.916777 {wncd_x_R0-0}{1}: [sisf-packet] [26328]: (debug): RX: ARP from interface cap
[...]
2021/09/28 12:59:52.810136 {wncd_x_R0-0}{1}: [client-iplearn] [26328]: (note): MAC: 3423.874c.6bf7

Client IP learn successful. Method: ARP IP: 172.16.21.153


2021/09/28 12:59:52.810185 {wncd_x_R0-0}{1}: [epm] [26328]: (info): [0000.0000.0000:unknown] HDL = 0x0 
2021/09/28 12:59:52.810404 {wncd_x_R0-0}{1}: [auth-mgr] [26328]: (info): [3423.874c.6bf7:capwap_9000000
2021/09/28 12:59:52.810794 {wncd_x_R0-0}{1}: [auth-mgr-feat_wireless] [26328]: (info): [0000.0000.0000:
2021/09/28 12:59:52.810863 {wncd_x_R0-0}{1}: [client-iplearn] [26328]: (info): MAC: 3423.874c.6bf7

IP-learn state transition: S_IPLEARN_IN_PROGRESS -> S_IPLEARN_COMPLETE
```

第3層身份驗證和重定向過程：

<#root>

```
2021/09/28 12:59:52.811141 {wncd_x_R0-0}{1}: [client-auth] [26328]: (note): MAC: 3423.874c.6bf7

L3 Authentication initiated. LWA


2021/09/28 12:59:52.811154 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7 Client 
2021/09/28 12:59:55.324550 {wncd_x_R0-0}{1}: [webauth-httpd] [26328]: (info): capwap_9000000b[3423.874c
2021/09/28 12:59:55.324565 {wncd_x_R0-0}{1}: [webauth-httpd] [26328]: (info): capwap_9000000b[3423.874c

HTTP GET request


2021/09/28 12:59:55.324588 {wncd_x_R0-0}{1}: [webauth-httpd] [26328]: (info): capwap_9000000b[3423.874c
[...]
2021/09/28 13:01:29.859434 {wncd_x_R0-0}{1}: [webauth-httpd] [26328]: (info): capwap_9000000b[3423.874c

POST rcvd when in LOGIN state


2021/09/28 13:01:29.859636 {wncd_x_R0-0}{1}: [webauth-acl] [26328]: (info): capwap_9000000b[3423.874c.6
2021/09/28 13:01:29.860335 {wncd_x_R0-0}{1}: [webauth-acl] [26328]: (info): capwap_9000000b[3423.874c.6
2021/09/28 13:01:29.861092 {wncd_x_R0-0}{1}: [auth-mgr] [26328]: (info): [3423.874c.6bf7:capwap_9000000

Authc success from WebAuth, Auth event success
```

```
2021/09/28 13:01:29.861151 {wncd_x_R0-0}{1}: [ewlc-infra-evq] [26328]: (note): Authentication Success.
2021/09/28 13:01:29.862867 {wncd_x_R0-0}{1}: [client-auth] [26328]: (note): MAC: 3423.874c.6bf7

L3 Authentication Successful.

 ACL:[]
2021/09/28 13:01:29.862871 {wncd_x_R0-0}{1}: [client-auth] [26328]: (info): MAC: 3423.874c.6bf7

Client auth-interface state transition: S_AUTHIF_WEBAUTH_PENDING -> S_AUTHIF_WEBAUTH_DONE
```

轉換為RUN狀態：

<#root>

```
2021/09/28 13:01:29.863176 {wncd_x_R0-0}{1}: [client-auth] [26328]: (note): MAC: 3423.874c.6bf7 ADD MOB
2021/09/28 13:01:29.863272 {wncd_x_R0-0}{1}: [errmsg] [26328]: (info): %CLIENT_ORCH_LOG-6-CLIENT_ADDED_

Username entry (3423.874C.6BF7) joined with ssid (EWA-Guest) for device with MAC: 3423.874c.6bf7


2021/09/28 13:01:29.863334 {wncd_x_R0-0}{1}: [aaa-attr-inf] [26328]: (info): [ Applied attribute :bsn-v
2021/09/28 13:01:29.863336 {wncd_x_R0-0}{1}: [aaa-attr-inf] [26328]: (info): [ Applied attribute : time
2021/09/28 13:01:29.863343 {wncd_x_R0-0}{1}: [aaa-attr-inf] [26328]: (info): [ Applied attribute : url-
2021/09/28 13:01:29.863387 {wncd_x_R0-0}{1}: [ewlc-qos-client] [26328]: (info): MAC: 3423.874c.6bf7 Cli
2021/09/28 13:01:29.863409 {wncd_x_R0-0}{1}: [rog-proxy-capwap] [26328]: (debug):

Managed client RUN state notification

: 3423.874c.6bf7
2021/09/28 13:01:29.863451 {wncd_x_R0-0}{1}: [client-orch-state] [26328]: (note): MAC: 3423.874c.6bf7

Client state transition: S_CO_L3_AUTH_IN_PROGRESS -> S_CO_RUN
```

關於此翻譯