Catalyst 9800 WLCでのローカルEAP認証の設定

内容

はじめに

前提条件

- 要件
- 使用するコンポーネント

設定

- ネットワーク図
- メインのローカルEAP設定
  - ステップ 1：ローカルEAPプロファイル
  - ステップ 2：AAA認証方式
  - ステップ 3：AAA許可方式の設定
  - ステップ 4：ローカルの高度な方式の設定
  - ステップ 5：WLAN の設定
  - 手順 6：1人以上のユーザを作成する
  - 手順 7：ポリシープロファイルを作成します。このWLANプロファイルをポリシープロファイルにマッピングするポリシータグを作成します。
  - ステップ 8：ポリシータグをアクセスポイントに導入します。

確認

トラブルシュート

- 誤ったパスワードが原因で接続に失敗するクライアントの例
- 障害時のトレース

はじめに

このドキュメントでは、Catalyst 9800 WLC（ワイヤレスLANコントローラ）でのローカルEAPの設定について説明します。

前提条件

要件

このドキュメントでは、Catalyst 9800 WLCでのローカルEAP(Extensible Authentication Protocol)の設定について説明します。つまり、WLCはワイヤレスクライアントに対してRADIUS認証サーバとして動作します。

このドキュメントでは、9800 WLCでのWLANの基本設定を十分に理解していることを前提に、ワイヤレスクライアント用のローカルEAPサーバとして動作するWLCのみを対象としています。

使用するコンポーネント

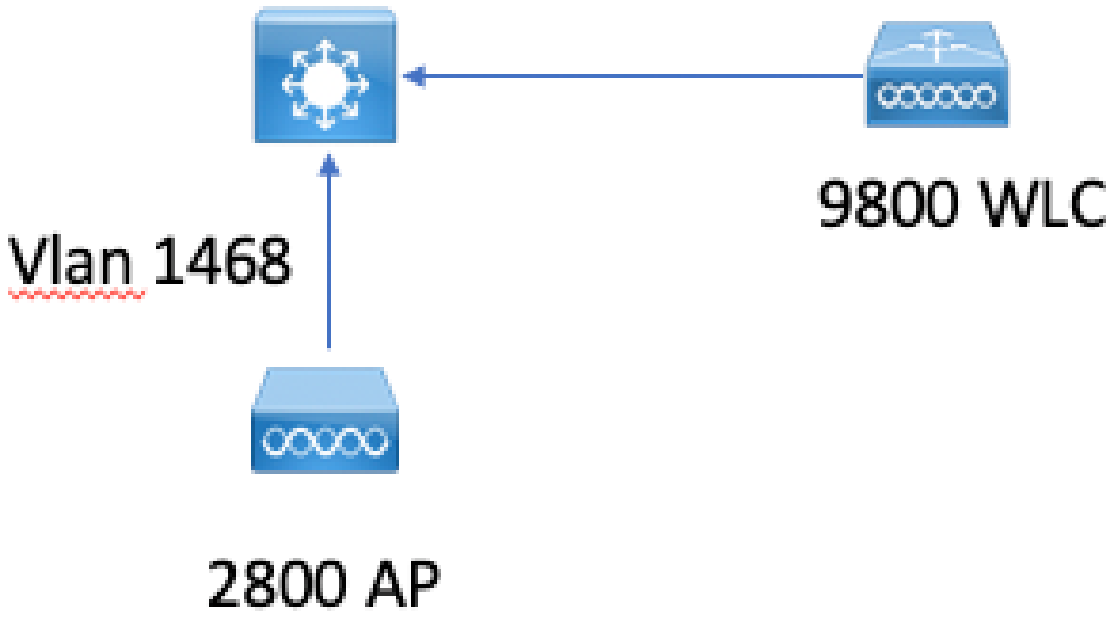
このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このド

キュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

バージョン16.12.1sのCatalyst 9800

設定

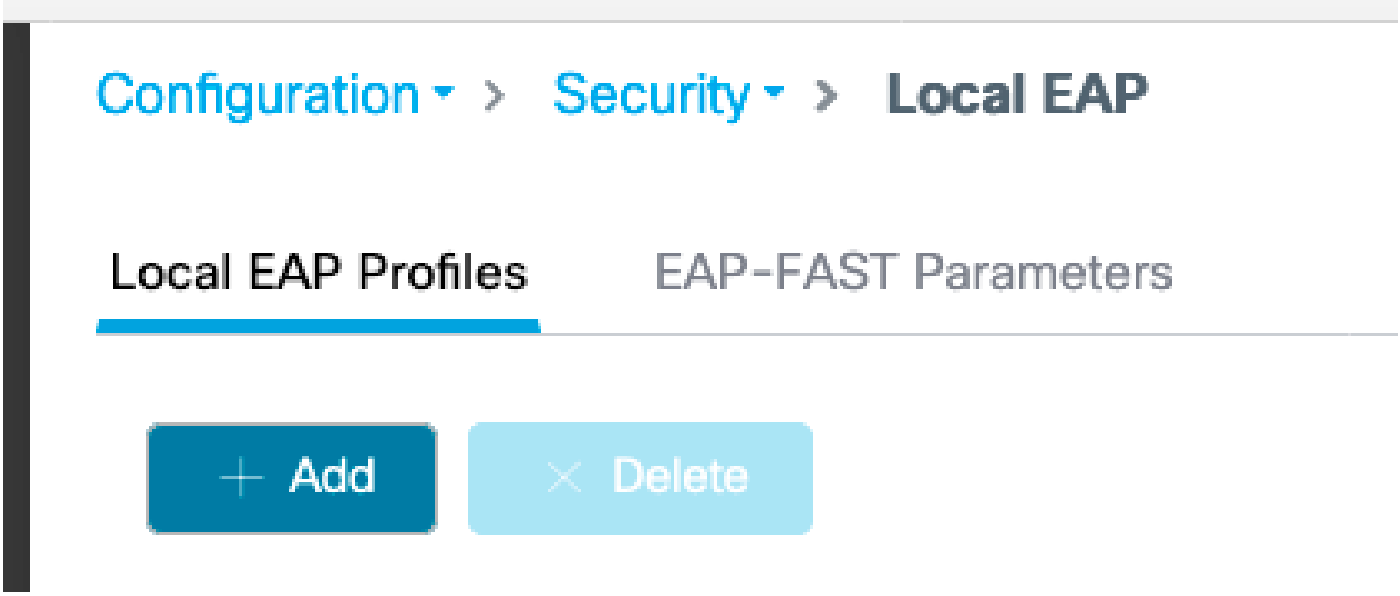
ネットワーク図



メインのローカルEAP設定

ステップ 1：ローカルEAPプロファイル

9800 Web UIでConfiguration > Security > Local EAPの順に選択します。

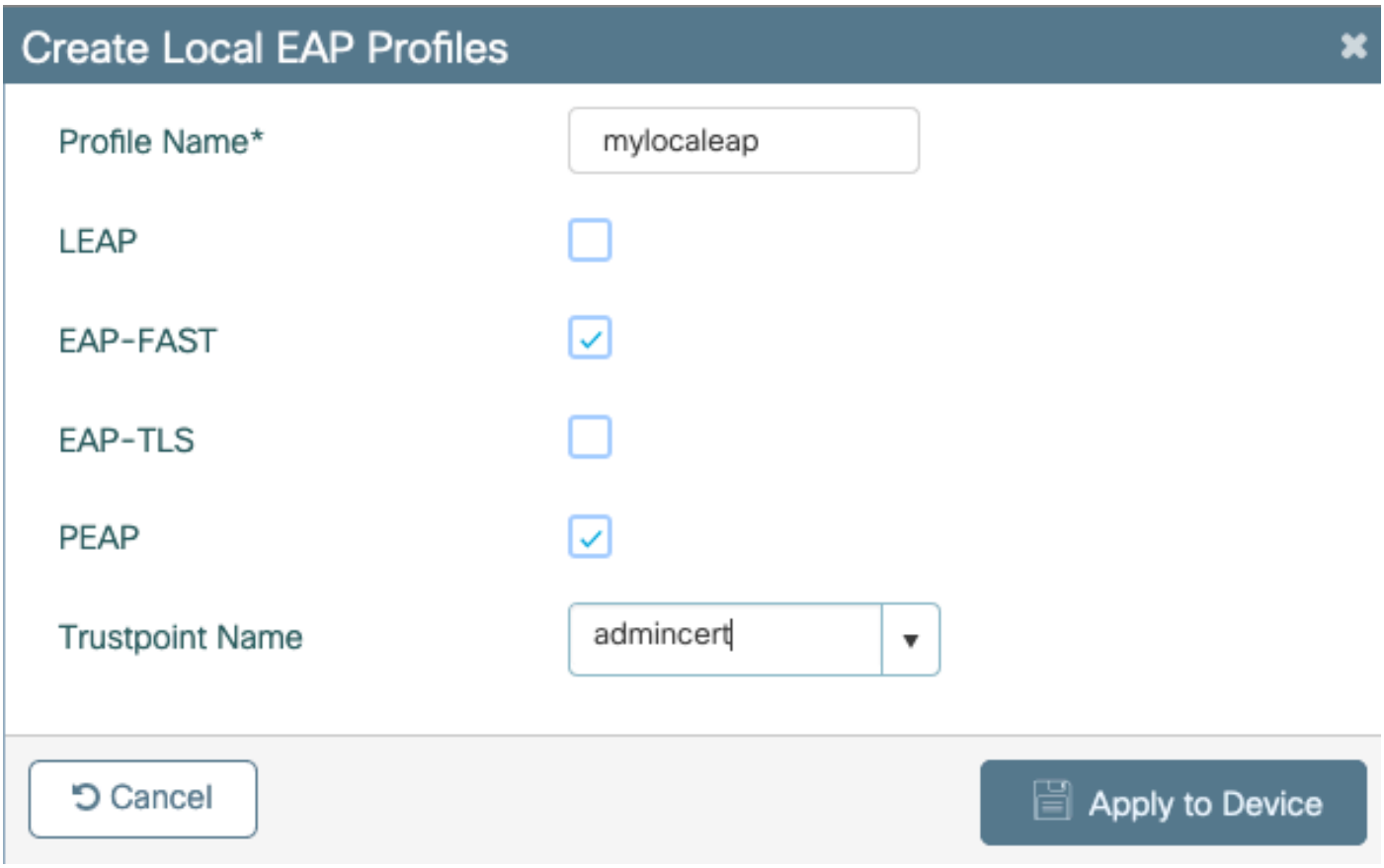


Addを選択します。

プロファイル名を入力します。

セキュリティが脆弱であるため、LEAPを使用することは推奨されません。他の3つのEAP方式では、トラストポイントを設定する必要があります。これは、オーセンティケータとして機能する9800は、クライアントが信頼できるように証明書を送信する必要があるためです。

クライアントはWLCのデフォルトの証明書を信頼しないため、クライアント側でサーバ証明書の検証を非アクティブにする（推奨されません）か、クライアントが信頼する9800 WLCに証明書トラストポイントをインストールする（またはクライアント信頼ストアに手動でインポートする）必要があります。



CLI：

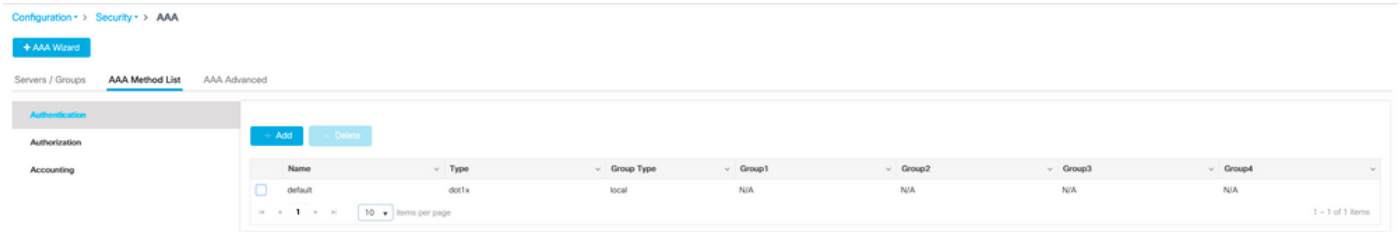
```
(config)#eap profile mylocapeap
(config-eap-profile)#method peap
(config-eap-profile)#pki-trustpoint admincert
```

ステップ 2：AAA認証方式

ユーザのローカルデータベースを使用するには、ローカルで参照するAAA dot1x方式を設定する必要があります（ただし、たとえば外部LDAPルックアップを使用できます）。

Configuration> Security > AAAの順に選択し、AAA method listタブでAuthenticationを選択します。Addを選択します。

「dot1x」タイプとローカルグループタイプを選択します。



ステップ 3：AAA許可方式の設定

Authorizationサブタブに移動し、credential-downloadタイプの新しい方式を作成して、ローカルをポイントします。

network認可タイプに対しても同じことを実行します

CLI：

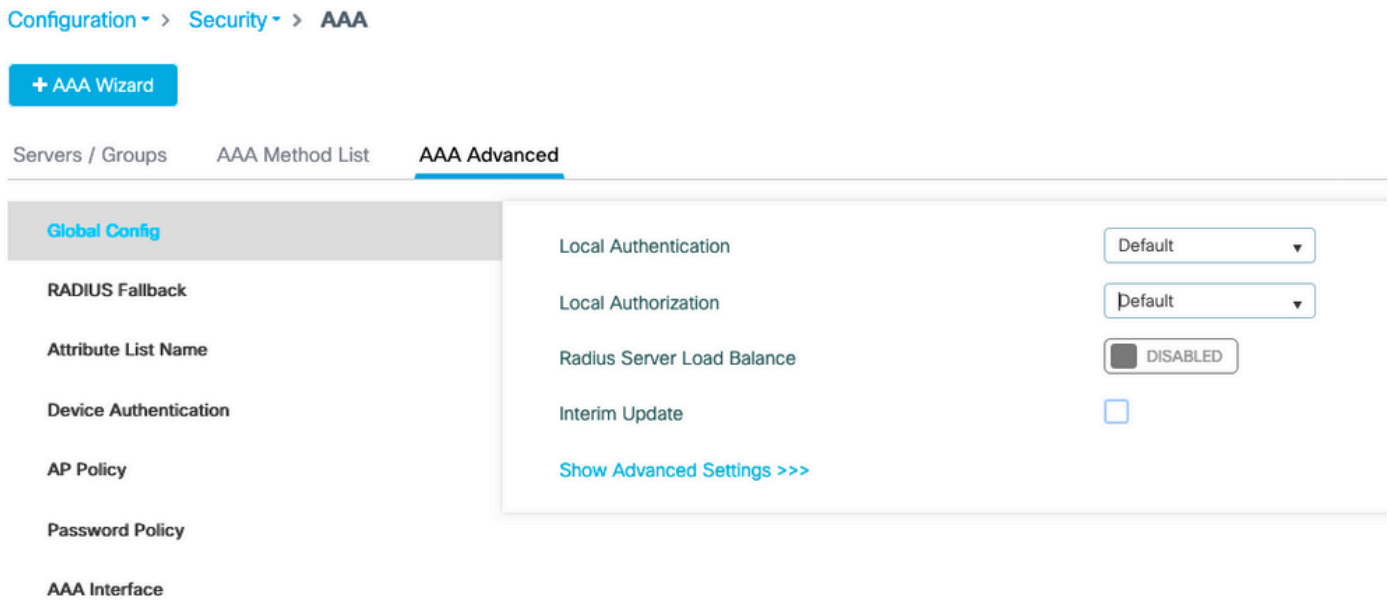
```
(config)#aaa new-model
(config)#aaa authentication dot1x default local
(config)#aaa authorization credential-download default local
(config)#aaa local authentication default authorization default
(config)#aaa authorization network default local
```

ステップ 4：ローカルの高度な方式の設定

AAA advancedタブに移動します。

ローカルの認証および許可方式を定義します。この例では「default」クレデンシャルダウンロード方式と「Default」dot1x方式を使用しているため、ここでローカル認証と許可の両方のドロップダウンボックスにデフォルトを設定する必要があります。

名前付きメソッドを定義した場合は、ドロップダウンから「メソッドリスト」を選択し、別のフィールドにメソッド名を入力します。



CLI：

```
aaa local authentication default authorization default
```

ステップ 5：WLAN の設定

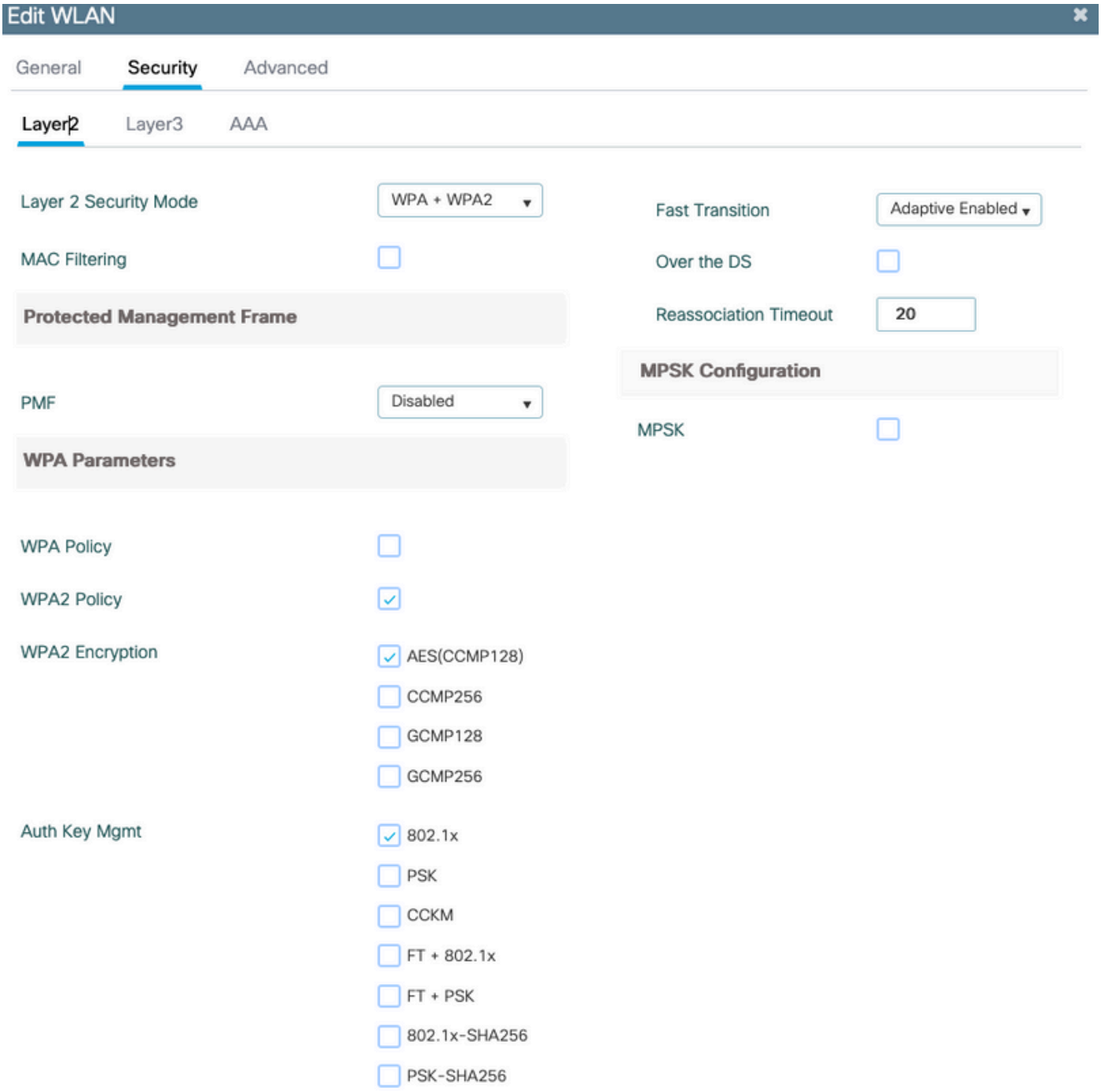
次に、前の手順で定義したローカルEAPプロファイルおよびAAA認証方式に対して、802.1xセキュリティ用のWLANを設定できます。

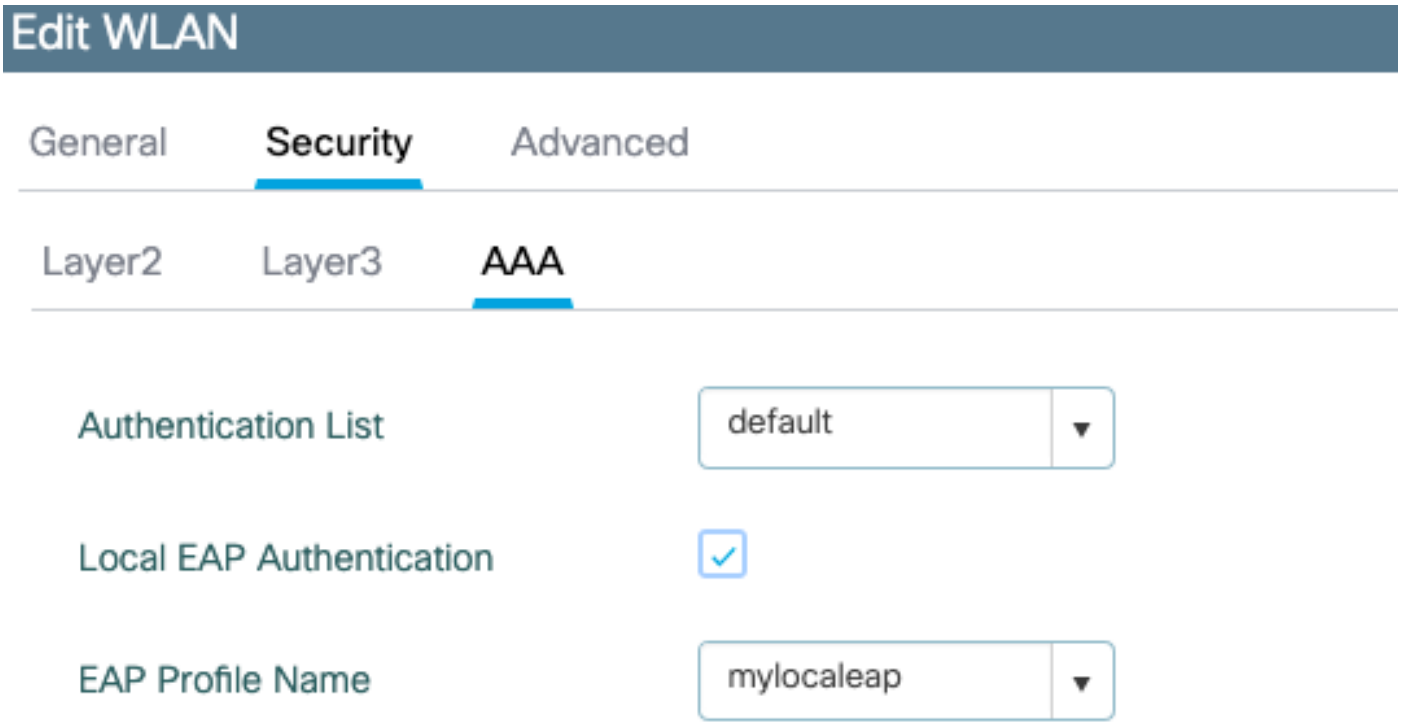
Configuration > Tags and Profiles > WLANs > + Addの順に選択します。

SSIDとプロファイル名を入力します。

Dot1xセキュリティは、レイヤ2の下でデフォルトで選択されています。

AAAの下で、Local EAP Authenticationを選択し、Local EAP profile and AAA Authentication list from drop-downを選択します。


Edit WLAN
General
Security
Advanced
Layer2
Layer3
AAA
Layer 2 Security Mode
WPA + WPA2
MAC Filtering
Protected Management Frame
PMF
Disabled
WPA Parameters
WPA Policy
WPA2 Policy
WPA2 Encryption
AES(CCMP128)
CCMP256
GCMP128
GCMP256
Auth Key Mgmt
802.1x
PSK
CCKM
FT + 802.1x
FT + PSK
802.1x-SHA256
PSK-SHA256
Fast Transition
Adaptive Enabled
Over the DS
Reassociation Timeout
20
MPSK Configuration
MPSK




```
(config)#wlan localpeapssid 1 localpeapssid
(config-wlan)#security dot1x authentication-list default
(config-wlan)#local-auth mylocaleap
```

手順 6：1人以上のユーザを作成する

CLIでは、ユーザはnetwork-userタイプである必要があります。CLIで作成されたユーザの例を次に示します。

```
(config)#user-name 1xuser
 creation-time 1572730075
 description 1xuser
 password 0 Cisco123
 type network-user description 1xuser
```

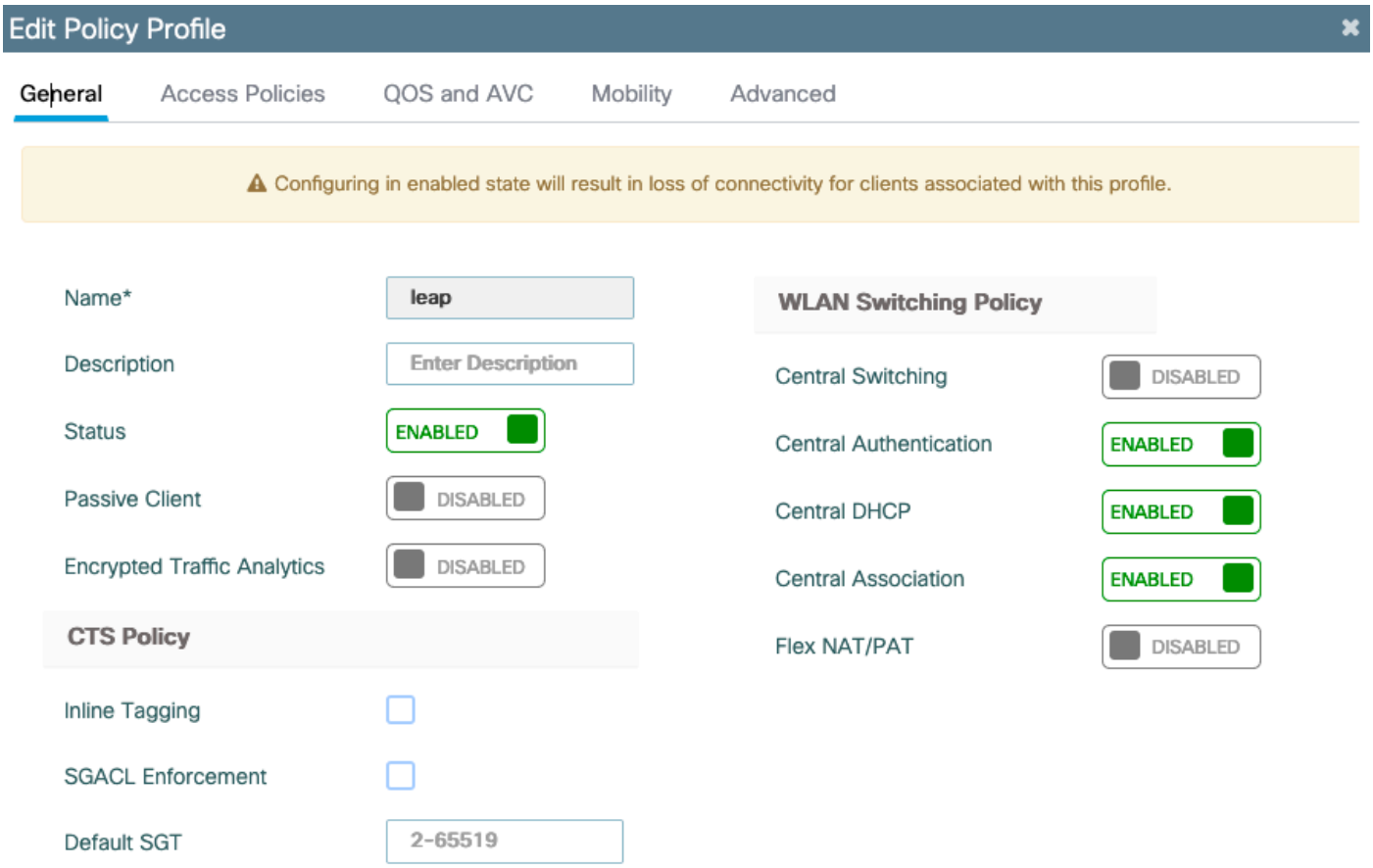
CLIで作成したユーザはWeb UIに表示されますが、Web UIで作成した場合、16.12の時点でnetwork-userにする方法はありません

手順 7：ポリシープロファイルを作成します。このWLANプロファイルをポリシープロファイルにマッピングするポリシータグを作成します。

Configuration > Tags and profiles > Policyの順に選択します。

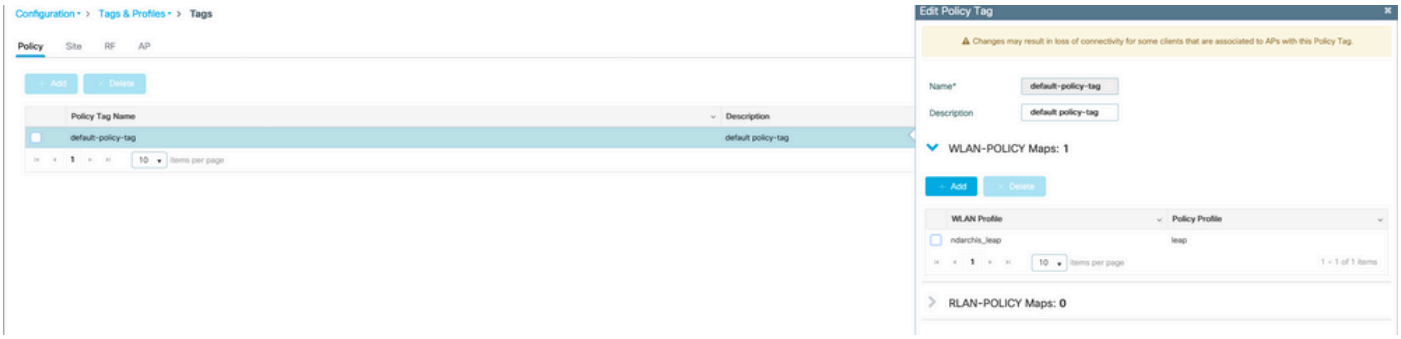
WLANのポリシープロファイルを作成します。

次の例は、vlan 1468でのflexconnectローカルスイッチングと中央認証のシナリオを示していますが、これはネットワークによって異なります。



Configuration > Tags and profiles > Tagsの順に選択します。

タグ内のポリシープロファイルにWLANを割り当てます。



ステップ 8：ポリシータグをアクセスポイントに導入します。

この場合、1つのAPに対して、AP上でタグを直接割り当てることができます。

Configuration > Wireless >Access pointsの順に進み、設定するAPを選択します。

割り当てたタグが設定したタグであることを確認します。

確認

主な設定行は次のとおりです。

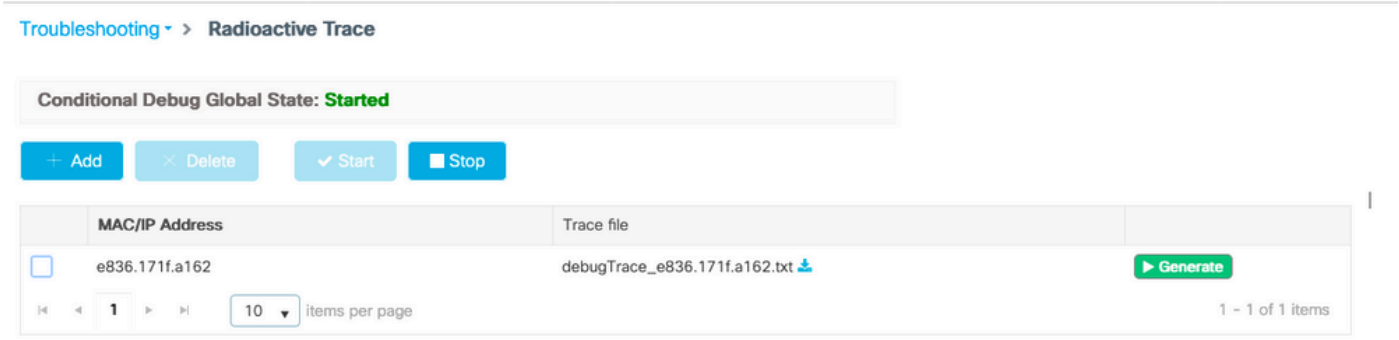
```
aaa new-model
aaa authentication dot1x default local
aaa authorization credential-download default local
aaa local authentication default authorization default
eap profile mylocaleap
method peap
pki-trustpoint admincert
user-name 1xuser
creation-time 1572730075 description 1xuser
password 0 Cisco123
type network-user description 1xuser
wlan ndarchis_leap 1 ndarchis_leap
local-auth mylocaleap
security dot1x authentication-list default
no shutdown
```

トラブルシュート

Cisco IOS® XE 16.12以前のリリースでは、ローカルEAP認証用にTLS 1.0のみがサポートされており、クライアントがTLS 1.2のみをサポートしている場合に問題が発生する可能性があります。これは、ますます一般的になっています。Cisco IOS® XE 17.1以降は、TLS 1.2およびTLS 1.0をサポートします。

接続に問題がある特定のクライアントをトラブルシューティングするには、RadioActive Tracingを使用します。Troubleshooting > RadioActive Traceの順に進み、クライアントのMACアドレスを追加します。

Startを選択して、そのクライアントのトレースを有効にします。



問題が再現されたら、Generateボタンを選択して、デバッグ出力を含むファイルを生成できます

。

誤ったパスワードが原因で接続に失敗するクライアントの例

```
2019/10/30 14:54:00.781 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.781 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.784 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.784 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.785 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.788 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.788 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.791 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.791 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.791 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.792 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.792 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.795 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.795 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.795 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.796 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.796 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.804 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.804 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.804 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.805 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.805 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [eap] [23294]: (info): FAST:EAP_FAIL from inner method MSCHAP
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Sen
2019/10/30 14:54:00.808 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.811 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rec
2019/10/30 14:54:00.811 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] EAP
2019/10/30 14:54:00.811 {wncd_x_R0-0}{2}: [caaa-authen] [23294]: (info): [CAAA:AUTHEN:66000006] DEBUG: r
2019/10/30 14:54:00.812 {wncd_x_R0-0}{2}: [eap-auth] [23294]: (info): FAIL for EAP method name: EAP-FAS
2019/10/30 14:54:00.812 {wncd_x_R0-0}{2}: [dot1x] [23294]: (info): [e836.171f.a162:capwap_90000004] Rai
2019/10/30 14:54:00.813 {wncd_x_R0-0}{2}: [errmsg] [23294]: (note): %DOT1X-5-FAIL: Authentication faile
2019/10/30 14:54:00.813 {wncd_x_R0-0}{2}: [auth-mgr] [23294]: (info): [e836.171f.a162:capwap_90000004] A
```

障害時のトレース

デバッグが有効になっていない場合でも、trace-on-failureコマンドを使用して、特定のMACアドレスの障害イベントのリストを確認できます。

次の例では、最初はAAA方式（AAAサーバダウンイベント）が存在せず、数分後にクライアントが誤ったクレデンシャルを使用しました。

コマンドは、リリース16.12以前ではshow logging trace-on-failure summaryであり、Cisco IOS® XE 17.1以降ではshow logging profile wireless (filter mac <mac>) trace-on-failureです。17.1以降では、クライアントのMACアドレスのフィルタリングが可能であるという点を除けば、技術的な違いはありません。

```
Nico9800#show logging profile wireless filter mac e836.171f.a162 trace-on-failure
Displaying logs from the last 0 days, 0 hours, 10 minutes, 0 seconds
executing cmd on chassis 2 ...
sending cmd to chassis 1 ...
Collecting files on current[1] chassis.
# of files collected = 30
Collecting files on current[2] chassis.
# of files collected = 30
Collecting files from chassis 1.
Time                          UUID                 Log

--------------------------------------------------------------------------------------------------


2019/10/30 14:51:04.438     0x0      SANET_AUTHC_FAILURE - AAA Server Down username , audit session id
2019/10/30 14:58:04.424     0x0      e836.171f.a162 CLIENT_STAGE_TIMEOUT State = AUTHENTICATING, WLAN p
```

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。