ユーザ アクセス、認証およびアカウンティング

この章の内容は、次のとおりです。

アクセス権のワークフローの依存関係

Cisco ACI RBAC のルールは、ファブリックの一部または全体へのアクセスを有効にするか、または制限します。たとえば、ベア メタル サーバ アクセス用のリーフ スイッチを設定するには、ログインしている管理者が `infra` ドメインに対する権限を持っている必要があります。デフォルトでは、テナント管理者は `infra` ドメインに対する権限を持っていません。この場合、リーフ スイッチに接続されているベア メタル サーバの使用を計画しているテナント管理者は、そのために必要なすべての手順を実行することはできません。テナント管理者は、`infra` ドメインに対する権限を持っているファブリック管理者と連携する必要があります。ファブリック管理者は、ACI リーフ

ユーザ アクセス、認証およびアカウンティング

スイッチに接続されたベア メタル サーバを使用するアプリケーション ポリシーを導入するためにテナント管理者が使用するスイッチ設定ポリシーをセットアップします。

ユーザ アクセス、認証およびアカウンティング

APIC ポリシーは、Cisco ACI ファブリックのアクセス、認証、およびアカウンティング（AAA）機能を管理します。ユーザ権限、ロールおよびドメインとアクセス権限の継承を組み合わせることにより、管理者は非常に細分化された方法で管理対象オブジェクト レベルで AAA 機能を設定することができます。これらの設定は、REST API、CLI、または GUI を使用して実行できます。

マルチテナントのサポート

コア APIC 内部データ アクセス コントロール システムにより、マルチテナント分離が提供され、テナント間での個人情報の漏洩が防止されます。読み取り/書き込みの制約により、テナントによる他のテナントの設定、統計情報、障害、またはイベント データの参照が防止されます。管理者によって読み取り権限が割り当てられない限り、テナントはファブリックの設定、ポリシー、統計情報、障害、またはイベントの読み取りが制限されます。

ユーザ アクセス：ロール、権限、セキュリティ ドメイン

APIC では、ロールベース アクセス コントロール（RBAC）を介してユーザのロールに従ってアクセスが提供されます。ACI ファブリック ユーザは、次に関連付けられています。

- ロールのセット
- ロールごとの権限タイプ：アクセスなし、読み取り専用、または読み取り/書き込み
- ユーザがアクセスできる管理情報ツリー（MIT）の一部を識別する 1 つ以上のセキュリティ ドメイン タグ

ACI ファブリックは、管理対象オブジェクト（MO）レベルでアクセス権限を管理します。権限は、システム内の特定の機能に対するアクセスを許可または制限する MO です。たとえば、ファブリック機器は権限ビットです。このビットは、物理ファブリックの機器に対応するすべてのオブジェクト上で APIC によって設定されます。

ロールは権限ビットの集合です。たとえば、「管理者」ロールが「ファブリック機器」と「テナント セキュリティ」に対する権限ビットに設定されていると、「管理者」ロールにはファブリックの機器とテナント セキュリティに対応するすべてのオブジェクトへのアクセス権があります。

セキュリティ ドメインは、ACI MIT オブジェクト階層の特定のサブツリーに関連付けられたタグです。たとえば、デフォルトのテナント「common」にはドメイン タグ `common` が付いています。同様に、特殊なドメイン タグ `all` の場合、MIT オブジェクト ツリー全体が含まれます。管理者は、MIT オブジェクト階層にカスタム ドメイン タグを割り当てることができます。たとえば、管理者は「solar」という名前のテナントにドメイン タグ「solar」を割り当てることができます。MIT 内では、特定のオブジェクトだけがセキュリティ ドメインとしてタグ付けできます。たとえば、

テナントはセキュリティ ドメインとしてタグ付けすることができますが、テナント内のオブジェクトはできません。

ユーザを作成してロールを割り当てても、アクセス権は有効になりません。1 つ以上のセキュリティ ドメインにそのユーザを割り当てることも必要です。デフォルトでは、ACI ファブリックには事前作成された以下の 2 つの特殊なドメインが含まれています。

- `All`：MIT 全体へのアクセスを許可
- `Infra`：ファブリック アクセス ポリシーなどの、ファブリック インフラストラクチャのオブジェクトおよびサブツリーへのアクセスを許可

（注） ユーザのクレデンシャルが許可しない管理対象オブジェクトの読み取り操作の場合、「DN/Class Unauthorized to read」ではなく「DN/Class Not Found」というエラーが返されます。ユーザのクレデンシャルが許可しない管理対象オブジェクトへの書き込み操作の場合、「HTTP 401 Unauthorized」というエラーが返されます。GUI では、ユーザのクレデンシャルが許可しないアクションの場合、表示されないか、またはグレー表示されます。

事前に定義された一連の管理対象オブジェクト クラスをドメインに関連付けることができます。これらのクラスがオーバーラップすることはできません。ドメインの関連付けをサポートするクラスの例：

- レイヤ 2 およびレイヤ 3 のネットワークで管理されたオブジェクト
- ネットワーク プロファイル（物理、レイヤ 2、レイヤ 3、管理など）
- Quality of Service（QoS）ポリシー

ドメインに関連付けることができるオブジェクトが作成されると、ユーザは、ユーザのアクセス権の範囲内でオブジェクトにドメインを割り当てる必要があります。ドメインの割り当てはいつでも変更できます。

仮想マシン管理（VMM）ドメインがセキュリティ ドメインとしてタグ付けされている場合、セキュリティ ドメイン内のユーザは、対応するタグ付き VMM ドメインにアクセスできます。たとえば、solar という名前のテナントに sun というセキュリティ ドメインのタグが付いており、VMM ドメインにも sun というセキュリティ ドメインのタグが付いている場合、solar テナント内のユーザは各自のアクセス権限に従って VMM ドメインにアクセスできます。

ローカル ユーザの設定

初期の設定スクリプトで、管理者アカウントが設定され、管理者はシステム起動時の唯一のユーザとなります。APIC は、きめ細かなロールベースのアクセス コントロール システムをサポートしており、そのシステムでは、権限が少ない管理者以外のユーザを含め、ユーザ アカウントをさまざまなロールで作成することができます。

GUI を使用したローカル ユーザの設定

（注） このタスク例のビデオを視聴するには、ビデオを掲載している Web ページ を参照してください。

はじめる前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- 必要に応じて、ユーザがアクセスするセキュリティ ドメインが定義されていること。たとえば、新しい使用アカウントがテナントにアクセスすることを制限する場合は、それに従ってテナント ドメインにタグ付けします。
- 以下を行うことができる APIC ユーザ アカウントを使用できること。
  - TACACS+ と TACACS+ プロバイダー グループの作成。
  - ターゲット セキュリティ ドメインでのローカル ユーザ アカウントの作成。ターゲット ドメインが `all` である場合、新しいローカル ユーザの作成に使用するログイン アカウントは、`all` にアクセスできるファブリック全体の管理者である必要があります。ターゲット ドメインがテナントである場合、新しいローカル ユーザの作成に使用するログイン アカウントは、ターゲット テナント ドメインに対する完全な読み取り/書き込みアクセス権を持つテナント管理者である必要があります。

手順の概要

1. メニュー バーで、[ADMIN] > [AAA] を選択します。
2. [Navigation] ペインで、[AAA Authentication] をクリックします。
3. [Work] ペインのデフォルトの [Authentication] フィールドで、[Realm] フィールドが [Local] と表示されていることを確認します。
4. [Navigation] ペインで、[Security Management] > [Local Users] を展開します。
5. [Navigation] ペインで、[Create Local User] を右クリックします。
6. [Security] ダイアログボックスで、ユーザに必要なセキュリティ ドメインを選択し、[Next] をクリックします。
7. [Roles] ダイアログボックスで、ユーザのロールを選択するためのオプション ボタンをクリックし、[Next] をクリックします。
8. [User Identity] ダイアログボックスで、次の操作を実行します。
9. [Navigation] ペインで、作成したユーザの名前をクリックします。[Work] ペインで、[Security Domains] 領域のユーザの横にある [+] 記号を展開します。

手順の詳細

ステップ 1 メニュー バーで、[ADMIN] > [AAA] を選択します。

ステップ 2 [Navigation] ペインで、[AAA Authentication] をクリックします。

ステップ 3 [Work] ペインのデフォルトの [Authentication] フィールドで、[Realm] フィールドが [Local] と表示されていることを確認します。

ステップ 4 [Navigation] ペインで、[Security Management] > [Local Users] を展開します。
管理ユーザはデフォルトで存在しています。

ステップ 5 [Navigation] ペインで、[Create Local User] を右クリックします。

ステップ 6 [Security] ダイアログボックスで、ユーザに必要なセキュリティ ドメインを選択し、[Next] をクリックします。

ステップ 7 [Roles] ダイアログボックスで、ユーザのロールを選択するためのオプション ボタンをクリックし、[Next] をクリックします。
読み取り専用または読み取り/書き込み権限を提供できます。

ステップ 8 [User Identity] ダイアログボックスで、次の操作を実行します。

a) [Login ID] フィールドで、ID を追加します。

b) [Password] フィールドにパスワードを入力します。
ユーザがパスワードを設定する時点で、APIC は以下の基準に対してパスワードを検証します。

- パスワードの最小長は 8 文字です。
- パスワードの最大長は 64 文字です。
- 連続して繰り返される文字は 3 文字未満です。
- 小文字、大文字、数字、記号の文字種のうち少なくとも 3 種類の文字が含まれている必要があります。
- 簡単に推測できるパスワードは使用しません。
- ユーザ名やユーザ名を逆にしたものは使用できません。
- cisco、isco、またはこれらの文字列の並べ替えを変化させたものや、それらの文字の大文字化の変更により取得される変形語であってはなりません。

c) [Confirm Password] フィールドで、パスワードを確認します。

d) [Finish] をクリックします。

ステップ 9 [Navigation] ペインで、作成したユーザの名前をクリックします。[Work] ペインで、[Security Domains] 領域のユーザの横にある [+] 記号を展開します。
ユーザのアクセス権限が表示されます。

リモート ユーザの設定

ローカルユーザを設定する代わりに、APICを一元化された企業クレデンシャルのデータセンターに向けることができます。APIC は、Lightweight Directory Access Protocol（LDAP）、Active Directory、RADIUS、および TACACS+ をサポートしています。

外部認証プロバイダーを通じて認証されたリモート ユーザを設定するには、次の前提条件を満たす必要があります。

- DNS 設定は、RADIUS サーバのホスト名ですでに名前解決されている必要があります。
- 管理サブネットを設定する必要があります。

外部認証サーバの AV ペア

Cisco 属性/値（AV）ペアを既存のユーザ レコードに追加して、ユーザ権限を APIC コントローラに伝播することができます。Cisco AV ペアは、APIC ユーザに対してロールベース アクセス コントロール（RBAC）のロールと権限を指定するために使用する単一の文字列です。オープンRADIUS サーバ（/etc/raddb/users）の設定例は次のとおりです。

```
aaa-network-admin Cleartext-Password := "<password>"
Cisco-avpair = "shell:domains = all/aaa/read-all(16001)"
```

AV ペアを割り当てるためのベスト プラクティス

ベストプラクティスとして、シスコは、bash シェルでユーザに割り当てられる AV ペアには 16000 ～ 23999 の範囲の一意の UNIX ユーザ ID を割り当てることを推奨します（SSH、Telnet または Serial/KVM のコンソールを使用）。Cisco AV ペアが UNIX ユーザ ID を提供しない状況が発生すると、そのユーザにはユーザ ID 23999 または範囲内の類似した番号が割り当てられます。これにより、そのユーザのホーム ディレクトリ、ファイル、およびプロセスに UNIX ID 23999 を持つリモート ユーザがアクセスできるようになってしまいます。

外部認証サーバの AV ペアの設定

属性/値（AV）のペア文字列のカッコ内の数字は、セキュア シェル（SSH）または Telnet を使用してログインしたユーザの UNIX ユーザ ID として使用されます。

手順の概要

1. 外部認証サーバの AV ペアの設定

手順の詳細

外部認証サーバの AV ペアの設定

Cisco AV ペアの定義は次のとおりです（シスコは、UNIX ユーザ ID が指定されているかどうかにかかわらず AV ペアをサポートします）

例：

```
* shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2
         * shell:domains =
domainA/writeRole1|writeRole2|writeRole3/readRole1|readRole2,domainB/writeRole1|writeRole2|writeRole3/readRole1|readRole2(8101)

These are the boost regexes supported by APIC:
uid_regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})(\\(\\d+\\))$");
regex("shell:domains\\s*[=:]\\s*((\\S+?/\\S*?/\\S*?),\\S+?/\\S*?/\\S*?){0,31})$");
```

次に、例を示します。

```
shell:domains = coke/tenant-admin/read-all,pepsi//read-all(16001)
```

TACACS+ アクセス用の APIC の設定

はじめる前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- TACACS+ サーバのホスト名または IP アドレス、ポート、およびキーを使用できること。
- APIC 管理 EPG を使用できること。
- 以下を行うことができる APIC ユーザ アカウントを使用できること。

◦ TACACS+ プロバイダーと TACACS+ プロバイダー グループの作成。

図 *1*：*TACACS+* プロバイダー

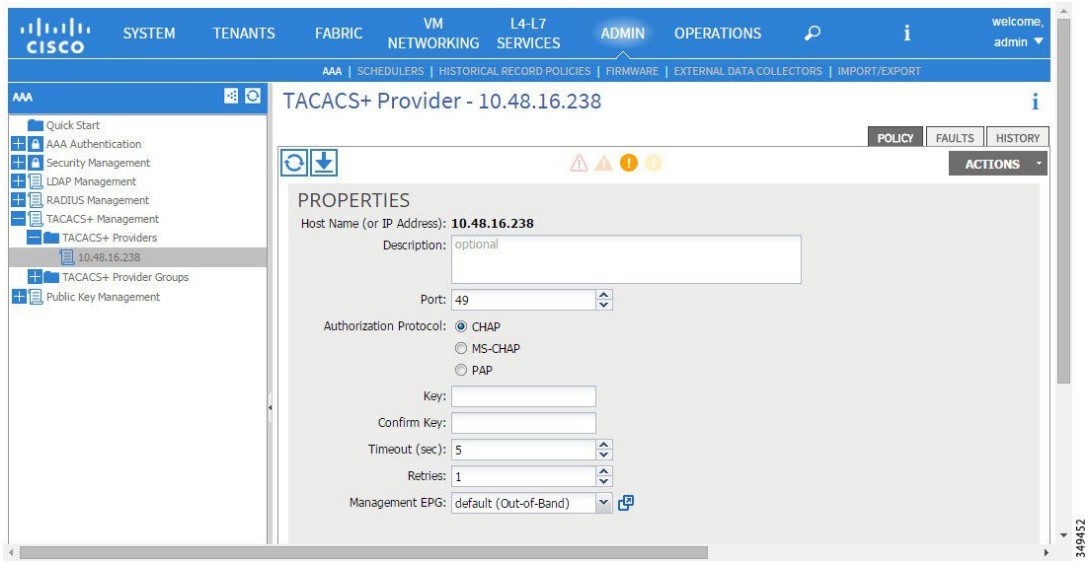
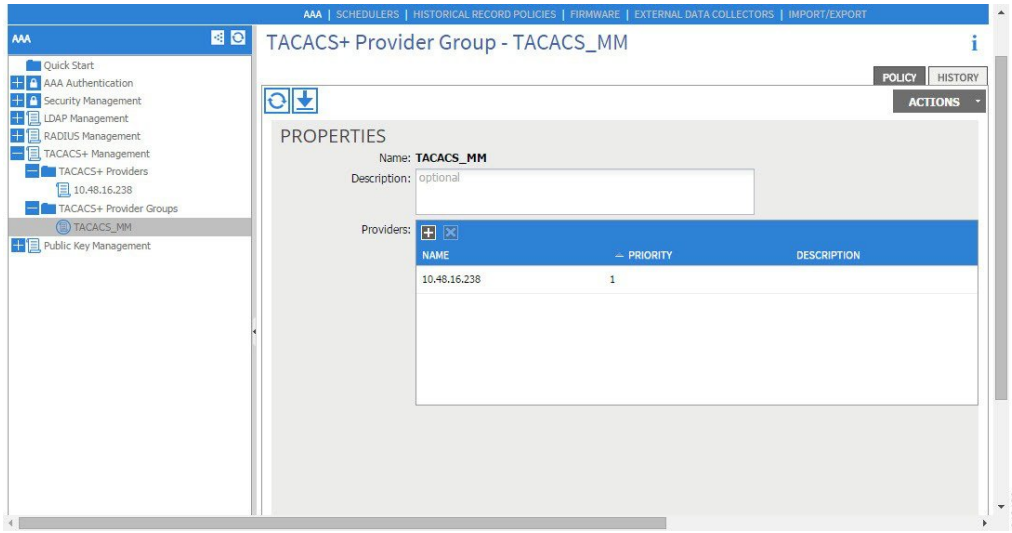
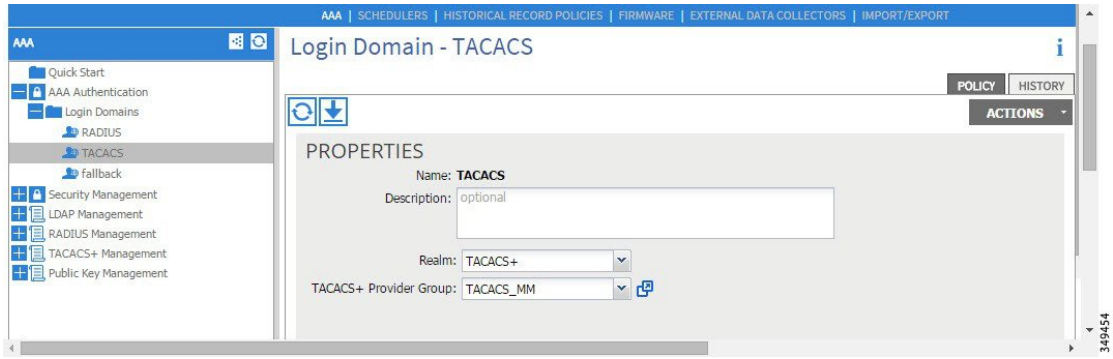


図 *2*：*TACACS+* プロバイダー グループ



- TACACS+ ログイン ドメインの作成。

図 3 : *TACACS+* の *AAA* ログイン ドメイン



ステップ 1 APIC で、[TACACS+ Provider] を作成します。

a) APIC メニュー バーで、[ADMIN] > [AAA] の順にクリックします。

b) [Navigation] ペインで、[+] アイコンをクリックして [TACACS+ Managment] オプションを展開します。

c) [Navigation] ペインで、[TACACS+ Providers] オプションを右クリックし、[Create TACACS+ Provider] を選択します。

d) TACACS+ ホスト名（または IP アドレス）、ポート、認証プロトコル、キー、および管理 EPG を指定します。

（注） APICがインバンド管理接続用に設定されている場合、TACACS+アクセス用にアウトオブバンド管理 EPG を選択しても有効にはなりません。また、インバンド管理 EPG 上のアウトオブバンドで TACACS+ サーバに接続することはできますが、TACACS+ サーバのスタティック ルートの設定が必要です。以下の Cisco ACS の設定手順例では、APIC インバンド IP アドレスを使用します。

ステップ 2 TACACS+ プロバイダー グループを作成します。

a) [Navigation] ペインで、[TACACS+ Provider Groups] オプションを右クリックし、[Create TACACS+ Provider Group] を選択します。

b) 必要に応じて、TACACS+ プロバイダー グループ名、説明、およびプロバイダーを指定します。

ステップ 3 TACACS+ の [Login Domain] を作成します。

a) [Navigation] ペインで、[+] アイコンをクリックして [AAA Authentication] オプションを展開します。

b) [Navigation] ペインで、[Login Domains] オプションを右クリックし、[Create Login Domain] を選択します。

c) 必要に応じて、ログイン ドメイン名、説明、レルム、およびプロバイダー グループを指定します。

次の作業

これで、APIC TACACS+ 設定手順は完了です。次に、RAIDUS サーバも使用する場合は、RADIUS 用の APIC の設定も行います。TACACS+ サーバのみを使用する場合は、次の ACS サーバ設定に関するトピックに進みます。

RADIUS アクセス用の APIC の設定

はじめる前に

- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- RADIUS サーバのホスト名または IP アドレス、ポート、認証プロトコル、およびキーを使用できること。
- APIC 管理 EPG を使用できること。
- 以下を行うことができる APIC ユーザ アカウントを使用できること。

- RADIUS プロバイダーと RADIUS プロバイダー グループの作成。

図 *4*：*RADIUS* プロバイダー

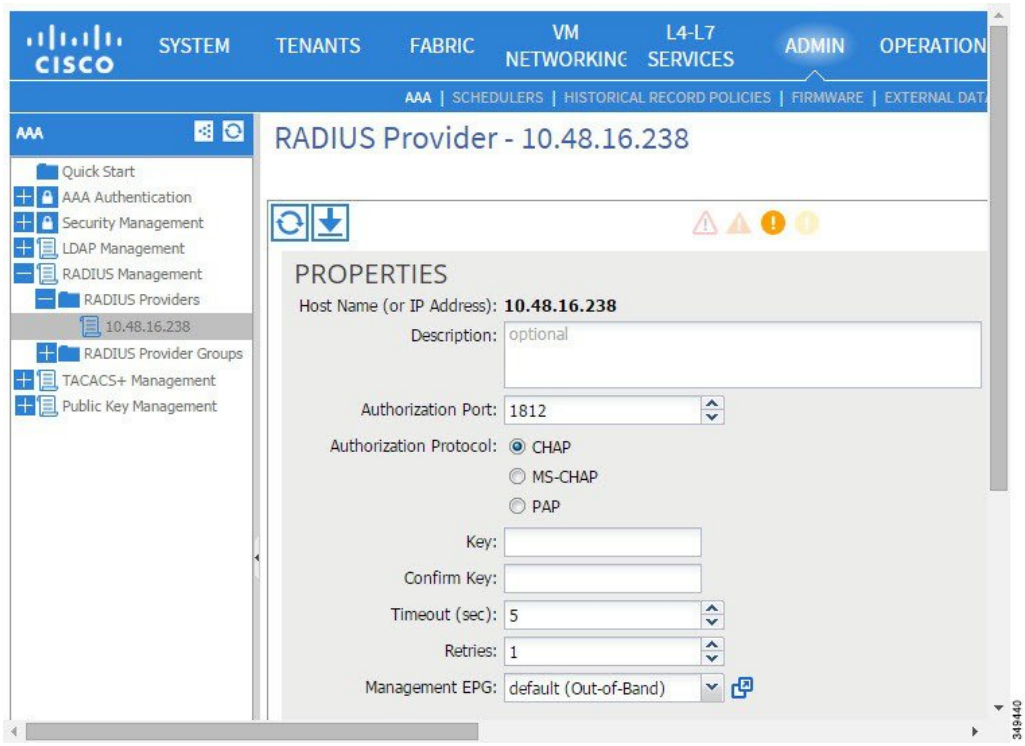
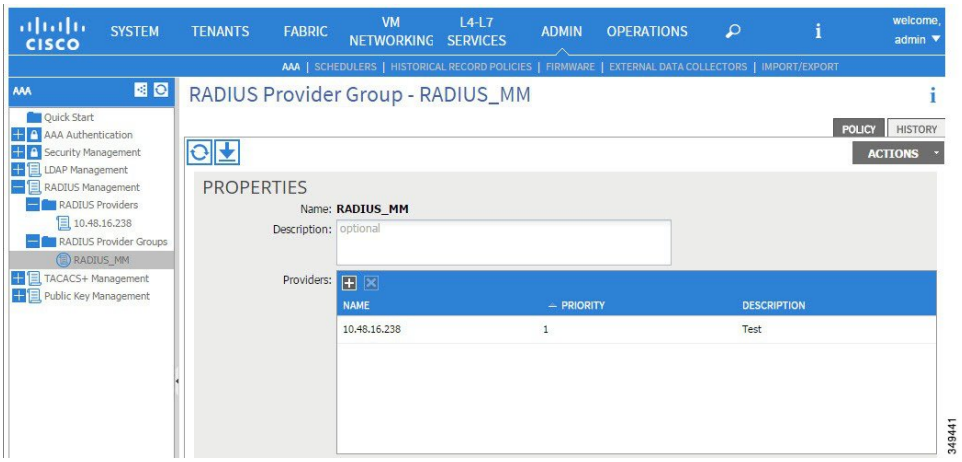
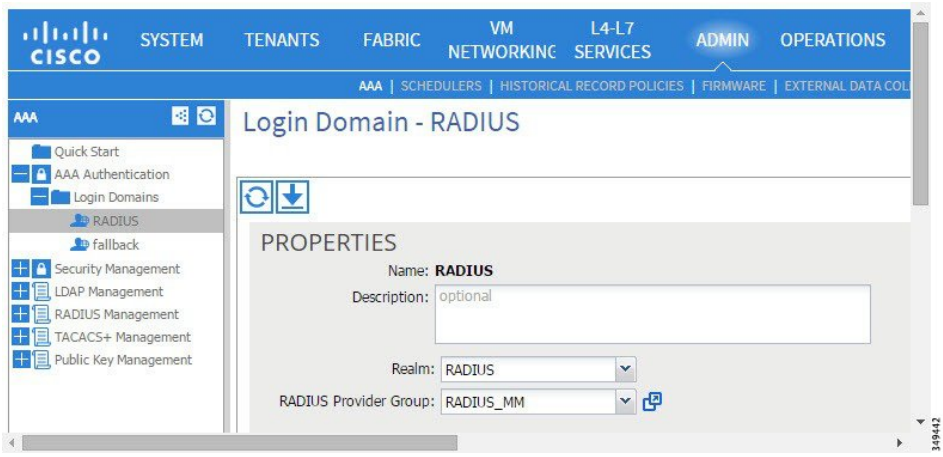


図 *5*：*RADIUS* プロバイダー グループ



- RADIUS ログイン ドメインの作成。

図 *6*： *RADIUS* の *AAA* ログイン ドメイン



ステップ 1 APIC で、[RADIUS Provider] を作成します。

a) APIC メニュー バーで、[ADMIN] > [AAA] の順にクリックします。

b) [Navigation] ペインで、[+] アイコンをクリックして [RADIUS Managment] オプションを展開します。

c) [Navigation] ペインで、[RADIUS Providers] オプションを右クリックし、[Create RADIUS Provider] を選択します。

d) RADIUS ホスト名（または IP アドレス）、ポート、プロトコル、および管理 EPG を指定します。

（注） APIC がインバンド管理接続用に設定されている場合、RADIUS アクセス用にアウトオブバンド管理 EPG を選択しても有効にはなりません。また、インバンド管理 EPG 上のアウトオブバンドで RADIUS サーバに接続することはできますが、RADIUS サーバのスタティック ルートの設定が必要です。以下の Cisco ACS の設定手順例では、APIC インバンド IP アドレスを使用します。

ステップ 2 [RADIUS Provider Group] を作成します。

a) [Navigation] ペインで、[RADIUS Provider Groups] オプションを右クリックし、[Create RADIUS Provider Group] を選択します。

b) 必要に応じて、RADIUS プロバイダー グループ名、説明、およびプロバイダーを指定します。

ステップ 3 RADIUS のログイン ドメインを作成します。

a) [Navigation] ペインで、[+] アイコンをクリックして [AAA Authentication] オプションを展開します。

b) [Navigation] ペインで、[Login Domains] オプションを右クリックし、[Create Login Domain] を選択します。

c) 必要に応じて、ログイン ドメイン名、説明、レルム、およびプロバイダー グループを指定します。

次の作業

これで、APIC RADIUS 設定手順は完了です。次に、RADIUS サーバを設定します。

APIC にアクセスする RADIUS および TACACS+ 用の Cisco Secure Access Control Server の設定

はじめる前に

- Cisco Secure Access Control Server（ACS）バージョン 5.5 がインストールされ、オンラインになっていること。
- APIC インバンド IP アドレスを使用できること。
- APIC の RADIUS キーまたは TACACS+ キーを使用できること（両方を設定する場合は両方のキー）。
- APIC コントローラがインストールされ、オンラインになっていること。APIC クラスタが形成されて正常に動作していること。
- RADIUS または TACACS+ のポート、認証プロトコル、およびキーを使用できること。
- 以下を行うことができる ACS ユーザ アカウントを使用できること。
  - APIC ACS クライアントの作成。

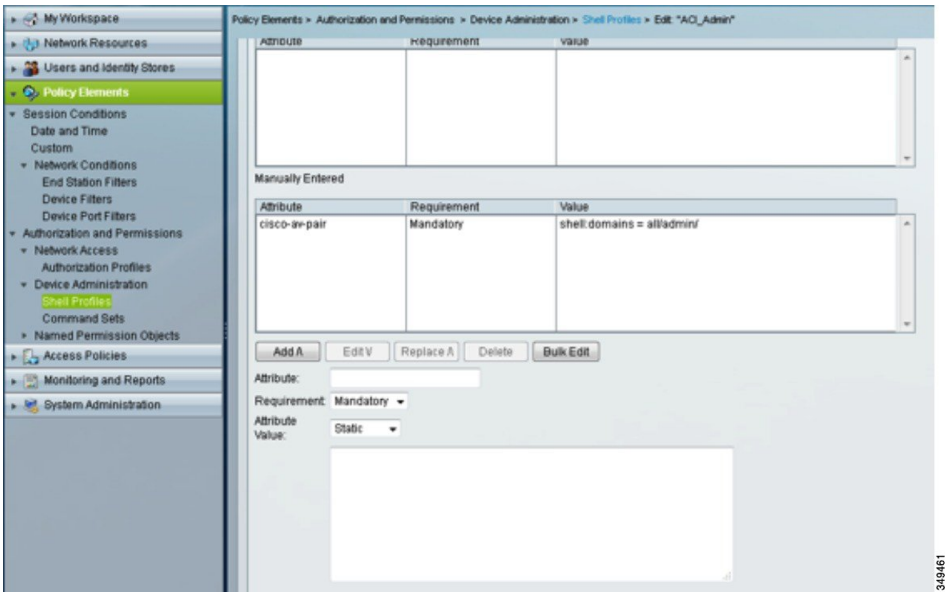
図 *7* : *APIC ACS* クライアント

- ユーザの定義と適切な ID グループへのマッピング。

図 *8*：*ID* グループへのユーザのマッピング

- APIC RBAC ロールを割り当てるための RADIUS ポリシー要素または TACACS+ シェル プロファイルの作成

図 *9*：*TACACS+* シェル プロファイル



ステップ 1　APIC をクライアントとして設定するには、ACS サーバにログインします。

a) [Network Resources] > [Network Devices Groups] > [Network Devices and AAA Clients] に移動します。
b) クライアント名と APIC インバンド IP アドレスを指定し、TACACS+ または RADIUS（または両方）の認証オプションを選択します。

（注） RADIUS または TACACS+ のみの認証が必要な場合は、必要なオプションのみを選択します。

c) 共有秘密（キー）や認証オプションに適したポートなど、認証の詳細を指定します。

（注） [Shared Secret] は APIC [Provider] キーと一致する必要があります。

ステップ 2 ID グループを作成します。

a) [Users and Identity Stores] > [Internal Groups] オプションに移動します。
b) 必要に応じて、[Name] と [Parent Group] を指定します。

ステップ 3 ユーザを ID グループにマッピングします。

a) [Navigation] ペインで、[Users and Identity Stores] > [Internal Identity Stores] > [Users] オプションをクリックします。
b) 必要に応じて、ユーザの [Name] と [Identity Group] を指定します。

ステップ 4 ポリシー要素を作成します。

a) [Policy Elements] オプションに移動します。
b) RADIUS の場合、[Authorization and Permissions] > [Network Access] > [Authorization Profiles Name] を指定します。TACACS+ の場合、必要に応じて、[Authorization and Permissions] > [Device Administration] > [Shell Profile Name] を指定します。
c) RADIUS の場合、必要に応じて、[Attribute] には「`cisco-av-pair`」、[Type] には「string」、[Value] には「shell:domains = <domain>/<role>/,<domain>// role」と指定します。TACACS+ の場合、必要に応じて、[Attribute] には「`cisco-av-pair`」、[Requirement] には「Mandatory」、[Value] には「shell:domains = <domain>/<role>/,<domain>// role」と指定します。
たとえば、*cisco-av-pair* が shell:domains = solar/admin/,common// read-all(16001) である場合、「solar」は ACI テナント、「admin」は solor というテナント内すべてに対する書き込み権限をこのユーザに付与するロール、「common」は ACI テナント common、「read-all(16001)」は ACI テナント common のすべてに対する読み取り権限をこのユーザに付与するロールです。

ステップ 5 サービス選択ルールを作成します。

a) RADIUS の場合、サービス選択ルールを作成して ID グループをポリシー要素に関連付けるには、[Access Policies] > [Default Device Network Access Identity] > [Authorization] に移動し、ルールの [Name]、[Status]、および [Conditions] を指定し、必要に応じて「`Internal Users:UserIdentityGroup in ALL Groups:<identity group name>`」を追加します。
b) TACACS+ の場合、サービス選択ルールを作成して ID グループをシェル プロファイルに関連付けるには、[Access Policies] > [Default Device Admin Identity] > [Authorization] に移動します。ルールの [Name] と [Conditions] を指定し、必要に応じて [Shell Profile] を選択します。

次の作業

新しく作成された RADIUS および TACACS+ のユーザを使用して、APIC にログインします。割り当てられた RBAC ロールと権限に従って正しい APIC セキュリティ ドメインにアクセスできることを確認します。ユーザは、明示的に許可されていない項目にアクセスできてはなりません。読み取り/書き込みアクセス権が、そのユーザに設定されたものと一致している必要があります。

APIC アクセス用の Windows Server 2008 LDAP の設定

はじめる前に

- 最初に LDAP サーバを設定し、次に APIC を LDAP アクセス用に設定します。
- Microsoft Windows Server 2008 がインストールされ、オンラインになっていること。
- Microsoft Windows Server 2008 サーバ マネージャの ADSI Edit ツールがインストールされていること。ADSI Edit をインストールするには、Windows Server 2008 サーバ マネージャのヘルプに記載されている手順に従ってください。
- `AciCiscoAVPair` の属性の指定：Common Name = AciCiscoAVPair、LDAP Display Name = AciCiscoAVPair、Unique X500 Object ID = 1.3.6.1.4.1.9.22.1、Description = AciCiscoAVPair、Syntax = Case Sensitive String。

（注） LDAP 設定のベスト プラクティスは、属性文字列として AciCiscoAVPair を使用することです。これにより、オブジェクト識別子（OID）の重複を許可しない一般的な LDAP サーバの制限に関連した問題が回避されます。つまり、ciscoAVPair OID がすでに使用されている場合です。

- 以下を行うことができる Microsoft Windows Server 2008 ユーザ アカウントを使用できること。
  - ADSI Edit を実行して `AciCiscoAVPair` 属性を Active Directory（AD）スキーマに追加する。
  - `AciCiscoAVPair` 属性に対するアクセス許可を持つように Active Directory LDAP ユーザを設定する。

ステップ 1 ドメイン管理者として Active Directory（AD）サーバにログインします。

ステップ 2 AD スキーマに `AciCiscoAVPair` 属性を追加します。

a) [Start] > [Run] に移動し、「mmc」と入力し、Enter を押します。
Microsoft Management Console（MMC）が開きます。

b) [File] > [Add/Remove Sanp-in] > [Add] に移動します。

c) [Add Standalonee Snap-in] ダイアログボックスで、[Active Directory Schema] を選択し、[Add] をクリックします。
MMC コンソールが開きます。
d) [Attributes] フォルダを右クリックし、[Create Attribute] オプションを選択します。
[Create New Attribute] ダイアログボックスが開きます。
e) [] に「AciCiscoAVPair」、[LDAP Display Name] に「AciCiscoAVPair」、[Unique X500 Object ID] に「1.3.6.1.4.1.9.22.1」と入力し、[Syntax] で「Case Sensitive String」を選択します。
f) [OK] をクリックして、属性を保存します。

ステップ 3 [User Properties] クラスを [CiscoAVPair] 属性が含まれるように更新します。

a) MMC コンソールで、[Classes] フォルダを展開し、[user] クラスを右クリックし、[Properties] を選択します。
[user Properties] ダイアログボックスが開きます。
b) [Attributes] タブをクリックし、[Optional] リストから「CiscoAVPair」を選択し、[Add] をクリックします。
[Select Schema Object] ダイアログボックスが開きます。
c) [Select a schema object:] リストで、「CiscoAVPair」を選択し、[Apply] をクリックします。
d) MMC コンソールで、[Active Directory Schema] を右クリックし、[Reload the Schema] を選択します。

ステップ 4 AciCiscoAVPair 属性のアクセス許可を設定します。
LDAP には AciCiscoAVPair 属性が含まれているため、LDAP ユーザに APICRBAC ロールを割り当てることにより APIC アクセス許可を付与する必要があります。

a) [ADSI Edit] ダイアログ ボックスで、APIC にアクセスする必要があるユーザを見つけます。
b) ユーザ名を右クリックし、[Properties] を選択します。
[<user> Properties] ダイアログボックスが開きます。
c) [Attribute Editor] タブをクリックし、「*AciCiscoAVPair*」属性を選択し、[Value] に「shell:domains = <domain>/<role>/,<domain>// role」と入力します。
たとえば、*AciCiscoAVPair* が shell:domains = solar/admin/,common// read-all(16001) である場合、「solar」は ACI テナント、「admin」は solor というテナント内すべてに対する書き込み権限をこのユーザに付与するロール、「common」は ACI テナント common、「read-all(16001)」は ACI テナント common のすべてに対する読み取り権限をこのユーザに付与するロールです。
d) [OK] をクリックして変更を保存し、[<user> Properties] ダイアログボックスを閉じます。

LDAP サーバは APIC にアクセスするように設定されます。

次の作業

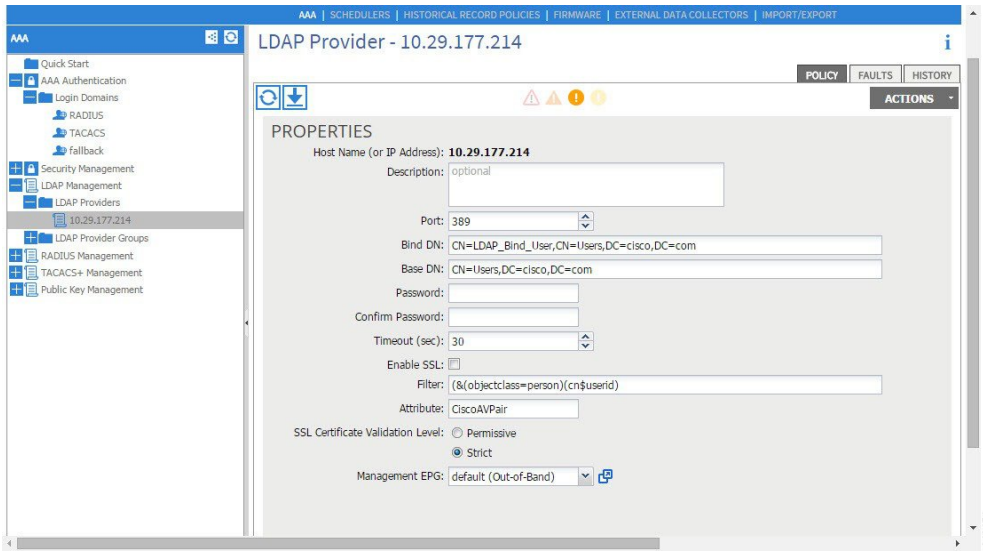
APIC を LDAP アクセス用に設定します。

LDAP アクセス用の APIC の設定

はじめる前に

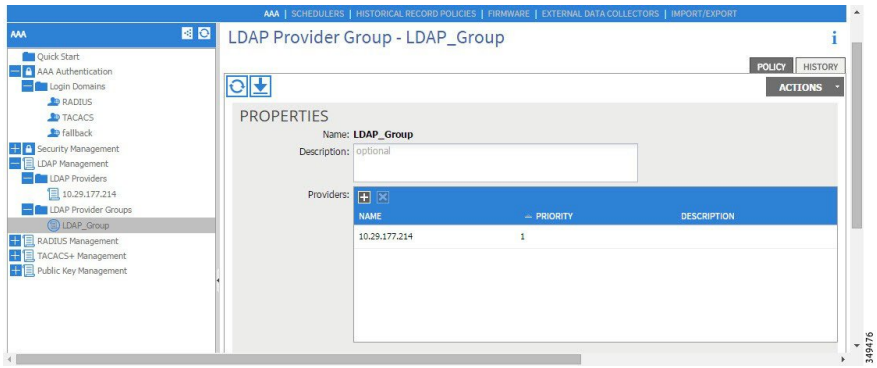
- ACI ファブリックが設置され、APIC コントローラがオンラインになっており、APIC クラスタが形成されて正常に動作していること。
- LDAP サーバのホスト名または IP アドレス、ポート、バインド DN、ベース DN、およびパスワードを使用できること。
- APIC 管理 EPG を使用できること。
- 以下を行うことができる APIC ユーザ アカウントを使用できること。
  - LDAP プロバイダーと LDAP プロバイダー グループの作成。

図 *10*： *LDAP* プロバイダー



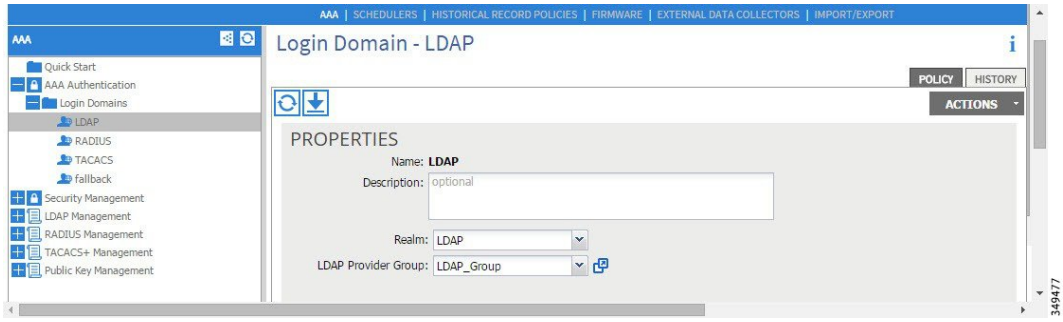
（注） バインド DN は、APIC が LDAP サーバにログインするために使用する文字列です。APICは、ログインしようとするリモートユーザを検証するためにこのアカウントを使用します。ベース DN は LDAP サーバのコンテナの名前とパスであり、そこで APIC がリモート ユーザ アカウントを検索します。これはパスワードが検証される場所です。フィルタを使用して、*cisco-av-pair* に使用することを APIC が 要求している属性を見つけます。これには、APIC で使用するユーザ認証と割り当て済み RBAC ロールが含まれます。APIC は、この属性を LDAP サーバに要求します。

図 *11*： *LDAP* プロバイダー グループ



◦ LDAP ログイン ドメインの作成。

図 *12*： *LDAP* の *AAA* ログイン ドメイン



ステップ 1 APIC で、LDAP プロバイダーを設定します。

a) APIC メニュー バーで、[ADMIN] > [AAA] の順にクリックします。

b) [Navigation] ペインで、[+] アイコンをクリックして [LDAP Managment] オプションを展開します。

c) [Navigation] ペインで、[LDAP Providers] オプションを右クリックし、[Create LDAP Provider] を選択します。

d) LDAP ホスト名（または IP アドレス）、ポート、バインド DN、ベース DN、パスワード、および管理 EPG を指定します。

（注） APIC がインバンド管理接続用に設定されている場合、LDAP アクセス用にアウトオブバンド管理 EPG を選択しても有効にはなりません。また、インバンド管理 EPG 上のアウトオブバンドで LDAP サーバに接続することはできますが、LDAP サーバのスタティック ルートの設定が必要です。本書の設定手順例では、APIC インバンド管理 EPG を使用します。

ステップ 2 APICで、LDAP プロバイダー グループを設定します。

a) [Navigation] ペインで、[LDAP Provider Groups] オプションを右クリックし、[Create LDAP Provider Group] を選択します。

b) 必要に応じて、LDAP プロバイダー グループ名、説明、およびプロバイダーを指定します。

ステップ 3 APIC で、LDAP のログイン ドメインを設定します。

a) [Navigation] ペインで、[+] アイコンをクリックして [AAA Authentication] オプションを展開します。

b) [Navigation] ペインで、[Login Domains] オプションを右クリックし、[Create Login Domain] を選択します。

c) 必要に応じて、ログイン ドメイン名、説明、レルム、およびプロバイダー グループを指定します。

次の作業

これで、APIC LDAP 設定手順は完了です。次に、APIC LDAP ログイン アクセスをテストします。

Cisco AV ペアが欠落しているか不良であるリモート ユーザのデフォルトの動作の変更

ステップ 1 メニュー バーで、[ADMIN] > [AAA] の順にクリックします。

ステップ 2 [Navigation] ペインで、[AAA Authentication] をクリックします。

ステップ 3 [Work] ペインの [Properties] 領域で、[Remote user login policy] ドロップダウン リストから、[Assign Default Role] を選択します。
デフォルト値は [No Login] です。[Assign Default Role] オプションは、Cisco AV ペアが欠落しているか不良であるユーザに最小限の読み取り専用権限を割り当てます。不正な AV ペアは、解析ルール適用時に問題があった AV ペアです。

署名ベースのトランザクションについて

Cisco ACI ファブリックの APIC コントローラは、ユーザを認証するためにさまざまな方法を提供します。

主要な認証方式ではユーザ名とパスワードが使用され、APIC REST API は APIC に対するその後のアクセスに使用できる認証トークンを返します。これは、HTTPS が使用不可であるか有効でない状況では安全でないと見なされます。

提供されている別の認証形式では、トランザクションごとに計算される署名が活用されます。その署名の計算には秘密キーが使用され、そのキーは安全な場所に保管して秘密にしておく必要があります。APIC がトークン以外の署名が付いた要求を受信すると、APIC は X.509 証明書を活用して署名を確認します。署名ベースの認証では、APIC に対するすべてのトランザクションに新しく計算された署名が必要です。これは、ユーザがトランザクションごとに手動で行うタスクではありません。理想的には、この機能は APIC と通信するスクリプトまたはアプリケーションで使用する必要があります。この方法では、攻撃者がユーザ クレデンシャルを偽装またはなりすますためには RSA/DSA キーを解読する必要があるため、最も安全です。

（注） また、リプレイ攻撃を防ぐためには HTTPS を使用する必要があります。

認証に X.509 証明書ベースの署名を使用する前に、次の必須タスクが完了していることを確認します。

1. OpenSSL または同様のツールを使用して X.509 証明書と秘密キーを作成します。
2. APIC のローカル ユーザを作成します（ローカル ユーザがすでに利用可能である場合、このタスクはオプションです）。
3. APIC のローカル ユーザに X.509 証明書を追加します。

注意事項と制約事項

次の注意事項と制約事項に従ってください。

- ローカル ユーザはサポートされます。リモート AAA ユーザはサポートされません。
- APIC GUI は証明書認証方式をサポートしません。
- WebSocket と eventchannel は X.509 要求では動作しません。
- サードパーティにより署名された証明書はサポートされません。自己署名証明書を使用します。

X.509 証明書と秘密キーの生成

ステップ 1 OpenSSL コマンドを入力して、X.509 証明書と秘密キーを生成します。

例：

```
$ openssl req -new -newkey rsa:1024 -days 36500 -nodes -x509 -keyout userabc.key -out userabc.crt
 -subj '/CN=User ABC/O=Cisco Systems/C=US'
```

（注）
- X.509 証明書が生成されると、APIC のユーザプロファイルに追加され、署名の確認に使用されます。秘密キーは、署名を生成するためにクライアントによって使用されます。
- 証明書には公開キーは含まれていますが、秘密キーは含まれていません。公開キーは、計算された署名を確認するために APIC によって使用される主要な情報です。秘密キーが APIC に保存されることはありません。このキーを秘密にしておく必要があります。

ステップ 2 OpenSSL を使用して証明書のフィールドを表示します。

例：

```
$ openssl x509 -text -in userabc.crt
        Certificate:
            Data:
                Version: 3 (0x2)
                Serial Number:
                    c4:27:6c:4d:69:7c:d2:b6
                Signature Algorithm: sha1WithRSAEncryption
                Issuer: CN=User ABC, O=Cisco Systems, C=US
                Validity
                    Not Before: Jan 12 16:36:14 2015 GMT
                    Not After : Dec 19 16:36:14 2114 GMT
                Subject: CN=User ABC, O=Cisco Systems, C=US
                Subject Public Key Info:
                    Public Key Algorithm: rsaEncryption
                    RSA Public Key: (1024 bit)
                        Modulus (1024 bit):
                            00:92:35:12:cd:2b:78:ef:9d:ca:0e:11:77:77:3a:
                            99:d3:25:42:94:b5:3e:8a:32:55:ce:e9:21:2a:ff:
                            e0:e4:22:58:6d:40:98:b1:0d:42:21:db:cd:44:26:
                            50:77:e5:fa:b6:10:57:d1:ec:95:e9:86:d7:3c:99:
                            ce:c4:7f:61:1d:3c:9e:ae:d8:88:be:80:a0:4a:90:
                            d2:22:e9:1b:25:27:cd:7d:f3:a5:8f:cf:16:a8:e1:
                            3a:3f:68:0b:9c:7c:cb:70:b9:c7:3f:e8:db:85:d8:
                            98:f6:e3:70:4e:47:e2:59:03:49:01:83:8e:50:4a:
                            5f:bc:35:d2:b1:07:be:ec:e1
                        Exponent: 65537 (0x10001)
                X509v3 extensions:
                    X509v3 Subject Key Identifier:
                        0B:E4:11:C7:23:46:10:4F:D1:10:4C:C1:58:C2:1E:18:E8:6D:85:34
                    X509v3 Authority Key Identifier:
                        keyid:0B:E4:11:C7:23:46:10:4F:D1:10:4C:C1:58:C2:1E:18:E8:6D:85:34
                        DirName:/CN=User ABC/O=Cisco Systems/C=US
                        serial:C4:27:6C:4D:69:7C:D2:B6

                    X509v3 Basic Constraints:
                        CA:TRUE
            Signature Algorithm: sha1WithRSAEncryption
                8f:c4:9f:84:06:30:59:0c:d2:8a:09:96:a2:69:3d:cf:ef:79:
                91:ea:cd:ae:80:16:df:16:31:3b:69:89:f7:5a:24:1f:fd:9f:
                d1:d9:b2:02:41:01:b9:e9:8d:da:a8:4c:1e:e5:9b:3e:1d:65:
                84:ff:e8:ad:55:3e:90:a0:a2:fb:3e:3e:ef:c2:11:3d:1b:e6:
                f4:5e:d2:92:e8:24:61:43:59:ec:ea:d2:bb:c9:9a:7a:04:91:
```

```
                8e:91:bb:9d:33:d4:28:b5:13:ce:dc:fe:c3:e5:33:97:5d:37:
                cc:5f:ad:af:5a:aa:f4:a3:a8:50:66:7d:f4:fb:78:72:9d:56:
                91:2c
        [snip]
```

ローカル ユーザの設定

GUI を使用したローカル ユーザの作成とユーザ証明書の追加

ステップ 1 メニュー バーで、[ADMIN] > [AAA] を選択します。

ステップ 2 [Navigation] ペインで、[AAA Authentication] をクリックします。

ステップ 3 [Work] ペインのデフォルトの [Authentication] フィールドで、[Realm] フィールドが [Local] と表示されていることを確認します。

ステップ 4 [Navigation] ペインで、[Security Management] > [Local Users] を展開します。
管理ユーザはデフォルトで存在しています。

ステップ 5 [Navigation] ペインで、[Local Users] をクリックし、[Create Local User] をクリックします。

ステップ 6 [Security] ダイアログボックスで、ユーザに必要なセキュリティ ドメインを選択し、[Next] をクリックします。

ステップ 7 [Roles] ダイアログボックスで、ユーザのロールを選択するためのオプション ボタンをクリックし、[Next] をクリックします。
読み取り専用または読み取り/書き込み権限を提供できます。

ステップ 8 [User Identity] ダイアログボックスで、次の操作を実行します。

a) [Login ID] フィールドで、ID を追加します。
b) [Password] フィールドにパスワードを入力します。
c) [Confirm Password] フィールドで、パスワードを確認します。
d) [Finish] をクリックします。

ステップ 9 [Navigation] ペインで、作成したユーザの名前をクリックします。[Work] ペインで、[Security Domains] 領域のユーザの横にある [+] 記号を展開します。
ユーザのアクセス権限が表示されます。

ステップ 10 [Work] ペインの [User Certificates] 領域で、ユーザ証明書の [+] 記号をクリックし、[Create X509 Certificate] ダイアログ ボックスで次の操作を実行します。

a) [Name] フィールドに、証明書の名前を入力します。
b) [Data] フィールドに、ユーザ証明書の詳細を入力します。
c) [Submit] をクリックします。

X509 証明書がローカル ユーザ用に作成されます。

REST API を使用したローカル ユーザの作成とユーザ証明書の追加

ローカル ユーザを作成し、ユーザ証明書を追加します。

例：

```
method: POST
        url: http://apic/api/node/mo/uni/userext/user-userabc.json
        payload:
        {
            "aaaUser": {
                "attributes": {
                    "name": "userabc",
                    "firstName": "Adam",
                    "lastName": "BC",
                    "phone": "408-525-4766",
                    "email": "userabc@cisco.com",
                },
                "children": [{
                    "aaaUserCert": {
                        "attributes": {
                            "name": "userabc.crt",
                         "data": "-----BEGIN CERTIFICATE-----\nMIICjjCCAfegAwIBAgIJAMQnbE <snipped
 content> ==\n-----END CERTIFICATE-----",
                        },
                        "children": []
                    },
                    "aaaUserDomain": {
                        "attributes": {
                            "name": "all",
                        },
                        "children": [{
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "aaa",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "access-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "admin",
                                    "privType": "writePriv",
                                },
                                "children": []
                            }
                        }, {
                            "aaaUserRole": {
                                "attributes": {
                                    "name": "fabric-admin",
                                    "privType": "writePriv",
                                },
                                "children": []
```

```
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "nw-svc-admin",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "ops",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "read-all",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "tenant-admin",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "tenant-ext-admin",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }, {
                    "aaaUserRole": {
                        "attributes": {
                            "name": "vmm-admin",
                            "privType": "writePriv",
                        },
                        "children": []
                    }
                }]
            }
        }]
    }
}
```

Python SDK を使用したローカル ユーザの作成

ローカル ユーザを作成します。

例：

```
#!/usr/bin/env python
from cobra.model.pol import Uni as PolUni
from cobra.model.aaa import UserEp as AaaUserEp
from cobra.model.aaa import User as AaaUser
from cobra.model.aaa import UserCert as AaaUserCert
from cobra.model.aaa import UserDomain as AaaUserDomain
from cobra.model.aaa import UserRole as AaaUserRole
from cobra.mit.access import MoDirectory
from cobra.mit.session import LoginSession
from cobra.internal.codec.jsoncodec import toJSONStr

APIC = 'http://10.10.10.1'
username = 'admin'
password = 'p@$$w0rd'

session = LoginSession(APIC, username, password)
modir = MoDirectory(session)
modir.login()

def readFile(fileName=None, mode="r"):
   if fileName is None:
      return ""
   fileData = ""
   with open(fileName, mode) as aFile:
      fileData = aFile.read()
   return fileData

# Use a dictionary to define the domain and a list of tuples to define
# our aaaUserRoles (roleName, privType)
# This can further be abstracted by doing a query to get the valid
# roles, that is what the GUI does

userRoles = {'all': [
               ('aaa', 'writePriv'),
               ('access-admin', 'writePriv'),
               ('admin', 'writePriv'),
               ('fabric-admin', 'writePriv'),
               ('nw-svc-admin', 'writePriv'),
               ('ops', 'writePriv'),
               ('read-all', 'writePriv'),
               ('tenant-admin', 'writePriv'),
               ('tenant-ext-admin', 'writePriv'),
               ('vmm-admin', 'writePriv'),
           ],
}

uni = PolUni('') # '' is the Dn string for topRoot
aaaUserEp = AaaUserEp(uni)
aaaUser = AaaUser(aaaUserEp, 'userabc', firstName='Adam',
                  email='userabc@cisco.com')

aaaUser.lastName = 'BC'
aaaUser.phone = '555-111-2222'
aaaUserCert = AaaUserCert(aaaUser, 'userabc.crt')
aaaUserCert.data = readFile("/tmp/userabc.crt")
# Now add each aaaUserRole to the aaaUserDomains which are added to the
# aaaUserCert
for domain,roles in userRoles.items():
   aaaUserDomain = AaaUserDomain(aaaUser, domain)
   for roleName, privType in roles:
       aaaUserRole = AaaUserRole(aaaUserDomain, roleName,
                                 privType=privType)
print toJSONStr(aaaUser, prettyPrint=True)

cr = ConfigRequest()
cr.addMo(aaaUser)
modir.commit(cr)
```

```
# End of Script to create a user
```

秘密キーを使用した署名の計算

はじめる前に

次の情報が用意されている必要があります。

- HTTP メソッド：GET、POST、DELETE
- 要求される REST API URI（クエリ オプションを含む）
- POST 要求の場合、APIC に送信される実際のペイロード
- ユーザの X.509 証明書の生成に使用される秘密キー
- APIC のユーザ X.509 証明書の宛先名

ステップ 1 HTTP メソッド、REST API URI、およびペイロードをこの順序で連結し、ファイルに保存します。OpenSSL で署名を計算するには、この連結データをファイルに保存する必要があります。この例では、ファイル名 payload.txt を使用します。秘密キーは userabc.key というファイルにあることに注意してください。

例：
GET の例：
```
GET http://10.10.10.1/api/class/fvTenant.json?rsp-subtree=children
```
POST の例：
```
POST http://10.10.10.1/api/mo/tn-test.json{"fvTenant": {"attributes": {"status": "deleted", "name":
 "test"}}}
```

ステップ 2 OpenSSL を使用して、秘密キーとペイロード ファイルを使用して署名を計算します。

例：
```
openssl dgst -sha256 -sign userabc.key payload.txt > payload_sig.bin
```
生成されたファイルには、複数行に印字された署名があります。

ステップ 3 Bash を使用して、署名から改行文字を取り除きます。

例：
```
$ tr -d '\n' < payload_sig.base64
P+OTqK0CeAZjl7+Gute2RlWw8OGgtzE0wsLlx8fIXXl4V79Zl7
Ou8IdJH9CB4W6CEvdICXqkv3KaQszCIC0+Bn07o3qF//BsIplZmYChD6gCX3f7q
IcjGX+R6HAqGeK7k97cNhXlWEoobFPe/oajtPjOu3tdOjhf/9ujG6Jv6Ro=
```
（注） これは、この特定の要求に関して APIC に送信される署名です。その他の要求では、独自の署名を計算する必要があります。

ステップ 4 署名を文字列内に配置し、APIC が署名をペイロードと照合して確認できるようにします。

この完全な署名が、要求のヘッダー内のクッキーとして APIC に送信されます。

例：

```
APIC-Request-Signature=P+OTqK0CeAZjl7+Gute2R1Ww8OGgtzE0wsLlx8f
IXXl4V79Zl7Ou8IdJH9CB4W6CEvdICXqkv3KaQszCIC0+Bn07o3qF//BsIplZmYChD6gCX3f
7qIcjGX+R6HAqGeK7k97cNhXlWEoobFPe/oajtPjOu3tdOjhf/9ujG6Jv6Ro=;
APIC-Certificate-Algorithm=v1.0; APIC-Certificate-Fingerprint=fingerprint;
APIC-Certificate-DN=uni/userext/user-userabc/usercert-userabc.crt
```

（注） ここで使用される DN が、次のステップの x509 証明書を含むユーザ認定オブジェクトの DN に一致する必要があります。

ステップ 5 署名を使用して APIC と通信するには、Python SDK の CertSession クラスを使用します。
次のスクリプトは、ACI Python SDK の CertSession クラスを使用して、署名を使用して APIC に要求する方法の例です。

例：

```
#!/usr/bin/env python
# It is assumed the user has the X.509 certificate already added to
# their local user configuration on the APIC
from cobra.mit.session import CertSession
from cobra.mit.access import MoDirectory

def readFile(fileName=None, mode="r"):
    if fileName is None:
        return ""
    fileData = ""
    with open(fileName, mode) as aFile:
        fileData = aFile.read()
    return fileData

pkey = readFile("/tmp/userabc.key")
csession = CertSession("https://ApicIPOrHostname/",
                       "uni/userext/user-userabc/usercert-userabc.crt", pkey)

modir = MoDirectory(csession)
resp = modir.lookupByDn('uni/fabric')
pring resp.dn
# End of script
```

（注） 前のステップで使用した DN が、このステップの x509 証明書を含むユーザ認定オブジェクトの DN に一致する必要があります。

アカウンティング

ACI ファブリック アカウンティングは、障害およびイベントと同じメカニズムで処理される以下の 2 つの管理対象オブジェクト（MO）によって処理されます。

- `aaaSessionLR` MO は、APIC およびスイッチでのユーザ アカウントのログイン/ログアウト セッション、およびトークン更新を追跡します。ACI ファブリック セッション アラート機能は、次のような情報を保存します。
  - ユーザ名

- セッションを開始した IP アドレス
- タイプ（telnet、https、REST など）
- セッションの時間と長さ
- トークン更新：ユーザ アカウントのログイン イベントは、ユーザ アカウントが ACI ファブリックの権利を行使するために必要な、有効なアクティブ トークンを生成します。

（注） トークンはログインに関係なく期限切れになります。ユーザはログアウトできますが、トークンは含まれているタイマー値の期間に従って期限切れになります。

- `aaaModLR` MO は、ユーザがオブジェクトに対して行う変更、およびいつ変更が発生したかを追跡します。

`aaaSessionLR` と `aaaModLR` 両方のイベント ログは、APIC シャードに保存されます。データがプリセットされているストレージ割り当てサイズを超えると、先入れ先出し方式でレコードを上書きします。

（注） APIC クラスタ ノードを破壊するディスク クラッシュや出火などの破壊的なイベントが発生した場合、イベント ログは失われ、イベント ログはクラスタ全体で複製されません。

`aaaModLR` MO と `aaaSessionLR` MO は、クラスまたは識別名（DN）でクエリできます。クラスのクエリは、ファブリック全体のすべてのログ レコードを提供します。ファブリック全体のすべての `aaaModLR` レコードは、GUI の [Fabric] -> [Inventory] -> [pod-1] -> [history] -> [audit log] セクションで取得できます。[GUI] => [History] => [Log] オプションを使用すると、GUI コンテキストで識別された特定のオブジェクトのイベント ログを表示できます。

標準の syslog、callhome、REST クエリ、および CLI エクスポート メカニズムは、aaaModLR MO と aaaSessionLR MO のクエリ データで完全にサポートされます。このデータをエクスポートするデフォルト ポリシーはありません。

APIC には、一連のオブジェクトまたはシステム全体のデータの集約を報告する、事前設定されたクエリはありません。ファブリック管理者は、`aaaModLR` および `aaaSessionLR` のクエリ データを定期的に syslog サーバにエクスポートするエクスポート ポリシーを設定できます。エクスポートされたデータを定期的にアーカイブし、システムの一部またはシステム ログ全体のカスタム レポートを生成するために使用できます。

共有サービスとしての外部ネットワークへのルーテッド接続の課金と統計情報

APIC は、共有サービスとしての外部ネットワークへのルーテッド接続用に設定されたポート（`l3extInstP` EPG）からバイト カウントとパケット カウントでの課金統計情報を収集するように設定できます。任意のテナントの任意の EPG が、外部ネットワークへのルーテッド接続用に `l3extInstP` EPG を共有できます。課金統計情報は、共有サービスとして `l3extInstP` EPG を使用する任意のテナント内の EPG ごとに収集できます。`l3extInstP` がプロビジョニングされているリーフ スイッチは課金統計情報を APIC に転送し、そこで課金情報が集約されます。定期的に課金統計情報をサーバにエクスポートするようにアカウンティング ポリシーを設定できます。