



## スイッチベースの認証の設定

---

この章では、Catalyst 3750 Metro スイッチでスイッチベースの認証を設定する方法について説明します。この章で説明する内容は、次のとおりです。

- [スイッチへの不正アクセスの防止 \(p.8-2\)](#)
- [イネーブル EXEC コマンドへのアクセスの保護 \(p.8-3\)](#)
- [TACACS+ によるスイッチ アクセスの制御 \(p.8-11\)](#)
- [RADIUS によるスイッチ アクセスの制御 \(p.8-19\)](#)
- [Kerberos によるスイッチ アクセスの制御 \(p.8-33\)](#)
- [スイッチのローカル認証および許可の設定 \(p.8-38\)](#)
- [SSH のためのスイッチの設定 \(p.8-39\)](#)
- [スイッチの SSL HTTP の設定 \(p.8-44\)](#)

## スイッチへの不正アクセスの防止

不正ユーザによる、スイッチの再設定や設定情報の閲覧を防止できます。一般的には、ネットワーク管理者からスイッチへのアクセスを許可する一方、非同期ポートを用いてネットワーク外からダイヤルアップ接続するユーザや、シリアルポートを通じてネットワーク外から接続するユーザ、またはローカル ネットワーク内の端末またはワークステーションから接続するユーザからのアクセスを制限します。

スイッチへの不正アクセスを防止するには、次のセキュリティ機能を 1 つまたは複数設定します。

- 最低限のセキュリティとして、各スイッチ ポートでパスワードおよび権限を設定します。このパスワードは、スイッチでローカルに保存されます。ユーザがポートまたは回線を通じてスイッチにアクセスしようとするとき、ポートまたは回線に指定されたパスワードを入力してからでなければ、スイッチにアクセスできません。詳細については、「[イネーブル EXEC コマンドへのアクセスの保護](#)」(p.8-3) を参照してください。
- 追加のセキュリティ レイヤとして、ユーザ名とパスワードをペアで設定できます。このペアはスイッチでローカルに保存されます。このペアは回線またはポートに割り当てられ、各ユーザを認証します。ユーザは認証後、スイッチにアクセスできます。権限レベルを定義している場合は、ユーザ名とパスワードの各ペアに特定の権限レベル (対応する権利および権限付き) を割り当てることもできます。詳細については、「[ユーザ名とパスワードのペアの設定](#)」(p.8-8) を参照してください。
- ユーザ名とパスワードのペアを使用したいが、そのペアをローカルではなく中央のサーバに保存したい場合は、セキュリティ サーバ上のデータベースに保存できます。これにより、複数のネットワーク デバイスが同じデータベースを使用してユーザ認証情報を (必要に応じて許可情報も) 得ることができます。詳細については、「[TACACS+ によるスイッチ アクセスの制御](#)」(p.8-11) を参照してください。

## イネーブル EXEC コマンドへのアクセスの保護

ネットワークで端末のアクセス制御を行う簡単な方法は、パスワードを使用して権限レベルを割り当てることです。パスワード保護によって、ネットワークまたはネットワーク デバイスへのアクセスが制限されます。権限レベルによって、ネットワーク デバイスにログオン後、ユーザがどのようなコマンドを入力できるかが定義されます。



(注)

ここで説明するコマンドの構文および使用方法の詳細については、『Cisco IOS Security Command Reference』 Release 12.2 を参照してください。

ここでは、コンフィギュレーション ファイルおよびイネーブル EXEC コマンドへのアクセスを制御する方法について説明します。具体的な設定情報は次のとおりです。

- デフォルトのパスワードおよび権限レベル設定 (p.8-3)
- スタティック イネーブル パスワードの設定または変更 (p.8-4)
- 暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護 (p.8-4)
- パスワード回復のディセーブル化 (p.8-6)
- 端末回線に対する Telnet パスワードの設定 (p.8-7)
- ユーザ名とパスワードのペアの設定 (p.8-8)
- 複数の権限レベルの設定 (p.8-8)

### デフォルトのパスワードおよび権限レベル設定

表 8-1 に、デフォルトのパスワードおよび権限レベル設定を示します。

表 8-1 デフォルトのパスワードおよび権限レベル

機能	デフォルト設定
イネーブル パスワードおよび権限レベル	パスワードは定義されていません。デフォルトはレベル 15 です (イネーブル EXEC レベル)。パスワードは、コンフィギュレーション ファイル内では暗号化されていない状態です。
イネーブル シークレット パスワードおよび権限レベル	パスワードは定義されていません。デフォルトはレベル 15 です (イネーブル EXEC レベル)。パスワードは、暗号化されてからコンフィギュレーション ファイルに書き込まれます。
回線パスワード	パスワードは定義されていません。

## スタティック イネーブル パスワードの設定または変更

イネーブル パスワードは、イネーブル EXEC モードへのアクセスを制御します。スタティック イネーブルパスワードを設定または変更するには、イネーブルEXECモードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>enable password <i>password</i></code>	イネーブルEXECモードのアクセス用に新しいパスワードを定義するか、既存のパスワードを変更します。  デフォルトでは、パスワードは定義されていません。  <i>password</i> には、1 ~ 25 文字の英数字のストリングを指定します。ストリングは数字で始めることはできません。大文字 / 小文字を区別し、スペースを使用できますが、先行スペースは無視されます。疑問符 (?) は、パスワードを作成する場合に、疑問符の前にキーの組み合わせ <code>Ctrl-v</code> を入力すれば使用できます。たとえば、パスワード <code>abc?123</code> を作成するときは、次のようにします。  <code>abc</code> を入力します。  <code>Ctrl-v</code> を入力します。  <code>?123</code> を入力します。  システムからイネーブル パスワードを入力するよう求められた場合、疑問符の前に <code>Ctrl-v</code> を入力する必要はなく、パスワードのプロンプトにそのまま <code>abc?123</code> と入力できます。
ステップ 3	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 4	<code>show running-config</code>	設定を確認します。
ステップ 5	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。  イネーブル パスワードは暗号化されず、スイッチのコンフィギュレーション ファイル内では読み取ることができる状態です。

パスワードを削除するには、`no enable password` グローバル コンフィギュレーション コマンドを使用します。

次に、イネーブルパスワードを `11u2c3k4y5` に変更する例を示します。パスワードは暗号化されておらず、レベル 15 のアクセスが与えられます (従来のイネーブル EXEC モードアクセス)。

```
Switch(config)# enable password 11u2c3k4y5
```


## 暗号化によるイネーブルおよびイネーブル シークレット パスワードの保護

特にネットワークを通過するパスワードや TFTP (簡易ファイル転送プロトコル) サーバに保存されているパスワードに対して、追加のセキュリティ レイヤを設定するには、`enable password` または `enable secret` グローバル コンフィギュレーション コマンドを使用できます。両コマンドはともに同じ働きをします。このコマンドにより、暗号化されたパスワードを設定できます。イネーブル EXEC モード (デフォルト設定) または特定の権限レベルにアクセスするには、このパスワードを入力する必要があります。

より高度な暗号化アルゴリズムを使用しているため、`enable secret` コマンドを使用することを推奨します。

**enable secret** コマンドは **enable password** コマンドに優先します。2 つのコマンドが同時に有効になることはありません。

イネーブル パスワードおよびイネーブル シークレット パスワードに暗号化を設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>enable password [level level] {password   encryption-type encrypted-password}</b>  または <b>enable secret [level level] {password   encryption-type encrypted-password}</b>	イネーブル EXEC モードのアクセス用に新しいパスワードを定義するか、既存のパスワードを変更します。  または シークレット パスワードを定義し、非可逆暗号方式を使用して保存します。  <ul style="list-style-type: none"> <li>• (任意) <i>level</i> に指定できる範囲は 0 ~ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。デフォルト レベルは 15 です (イネーブル EXEC モード権限)。</li> <li>• <i>password</i> には、1 ~ 25 文字の英数字のストリングを指定します。ストリングは数字で始めることはできません。大文字 / 小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されていません。</li> <li>• (任意) <i>encryption-type</i> には、シスコ独自の暗号化アルゴリズムであるタイプ 5 しか使用できません。暗号化タイプを指定する場合は、暗号化されたパスワードを使用する必要があります。この暗号化パスワードは、別のスイッチ コンフィギュレーションからコピーしたものです。</li> </ul> <p> (注) 暗号化タイプを指定してクリア テキスト パスワードを入力した場合は、再度イネーブル EXEC モードを開始することはできません。暗号化されたパスワードが失われた場合は、どのような方法でも回復することはできません。</p>
ステップ 3	<b>service password-encryption</b>	(任意) パスワードを定義するとき、またはコンフィギュレーションを保存するときに、パスワードを暗号化します。  暗号化によって、コンフィギュレーション ファイルのパスワードが読み取り不能になります。
ステップ 4	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 5	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

イネーブル パスワードおよびイネーブル シークレット パスワードの両方が定義されている場合、ユーザはイネーブル シークレット パスワードを入力する必要があります。

特定の権限レベルのパスワードを定義するには、**level** キーワードを使用します。レベルを指定してパスワードを設定したら、そのレベルでアクセスする必要のあるユーザだけにそのパスワードを渡してください。さまざまなレベルでアクセス可能なコマンドを指定する場合は、**privilege level** グローバル コンフィギュレーション コマンドを使用します。詳細については、「[複数の権限レベルの設定](#)」(p.8-8) を参照してください。

パスワードの暗号化をイネーブルにすると、ユーザ名パスワード、認証鍵パスワード、イネーブルコマンドパスワード、コンソールおよび仮想端末回線パスワードなど、すべてのパスワードに適用されます。

パスワードとレベルを削除するには、**no enable password [level level]** または **no enable secret [level level]** グローバル コンフィギュレーション コマンドを使用します。パスワードの暗号化をディセーブルにするには、**no service password-encryption** グローバル コンフィギュレーション コマンドを使用します。

次に、権限レベル2に対して暗号化パスワード \$1\$FaD0\$Xyti5Rkls3LoyxzS8 を設定する例を示します。

```
Switch(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

## パスワード回復のディセーブル化

デフォルトでは、スイッチに物理アクセスするすべてのエンドユーザは、スイッチの電源投入時にブートプロセスを中断して新しいパスワードを入力することで、失われたパスワードを回復できます。

パスワード回復ディセーブル機能では、この機能の一部をディセーブルにすることにより、スイッチパスワードへのアクセスを防止します。この機能がイネーブルな場合、エンドユーザはシステムをデフォルト設定に戻すことに同意するだけで、ブートプロセスを中断できます。パスワード回復をディセーブル化しても、ブートプロセスを中断してパスワードを変更できますが、コンフィギュレーションファイル (config.txt) および VLAN (仮想 LAN) データベースファイル (vlan.dat) は削除されます。



(注)

パスワード回復をディセーブルにする場合は、エンドユーザがブートプロセスを中断してシステムをデフォルト値に戻す場合に備えて、セキュアサーバ上にコンフィギュレーションファイルのバックアップコピーを保管しておいてください。スイッチ上にコンフィギュレーションファイルのバックアップコピーを保管しないでください。スイッチが VTP (VLAN トランク プロトコル) トランスペアレントモードで動作している場合は、セキュアサーバ上に VLAN データベースファイルのバックアップコピーも保管することを推奨します。スイッチがシステムのデフォルト設定に戻ると、Xmodem プロトコルを使用して、保存したファイルをダウンロードできます。詳細については、「[パスワードを忘れた場合の回復](#)」(p.40-4) を参照してください。

パスワード回復をディセーブルにするには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>no service password-recovery</b>	パスワード回復をディセーブルにします。  この設定は、フラッシュメモリのブートローダがアクセスできる領域および Cisco IOS イメージに保存されます。ただし、ファイルシステムの領域ではないのでユーザはアクセスできません。
ステップ 3	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 4	<b>show version</b>	コマンド出力の最後の数行を調べて設定を確認します。

パスワード回復を再びイネーブルにするには、**service password-recovery** グローバル コンフィギュレーション コマンドを使用します。



(注) **boot manual** グローバル コンフィギュレーション コマンドを使用して、スイッチを手動で起動するように設定している場合は、パスワード回復をディセーブルにできません。このコマンドにより、スイッチのオフ/オン後に、ブート ロード プロンプト (*switch:*) が表示されます。

## 端末回線に対する Telnet パスワードの設定

初めてスイッチに電源を投入すると、自動セットアッププログラムが起動して IP 情報を割り当て、引き続き使用するためのデフォルト設定を作成します。セットアッププログラムは、パスワードによる Telnet アクセス用にスイッチを設定するよう求めてきます。このとき、セットアッププログラムを使用してパスワードを設定しなかった場合は、CLI (コマンドライン インターフェイス) を使用して設定できます。

スイッチを Telnet アクセス用に設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1		エミュレーション ソフトウェアを備えた PC またはワークステーションと、スイッチのコンソール ポートを接続します。  コンソール ポートのデフォルトのデータ特性は、9600 ボー、8 データ ビット、1 ストップ ビット、パリティなしです。コマンドライン プロンプトを表示させるため、Return キーを数回押すこともあります。
ステップ 2	<b>enable password</b> <i>password</i>	イネーブル EXEC モードを開始します。
ステップ 3	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 4	<b>line vty 0 15</b>	Telnet セッション (回線) の数を設定し、ライン コンフィギュレーション モードを開始します。  コマンド対応スイッチでは、最大 16 のセッションが可能です。0 および 15 を指定することは、使用できる 16 個の Telnet セッションを全部設定することを意味します。
ステップ 5	<b>password</b> <i>password</i>	1 つまたは複数の回線の Telnet パスワードを入力します。  <i>password</i> には、1 ~ 25 文字の英数字のストリングを指定します。ストリングは数字で始めることはできません。大文字 / 小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されていません。
ステップ 6	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 7	<b>show running-config</b>	設定を確認します。  コマンド <b>line vty 0 15</b> の下にパスワードが表示されます。
ステップ 8	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

パスワードを削除するには、**no password** グローバル コンフィギュレーション コマンドを使用します。

Telnet パスワードを *let45me67in89* に設定する例を示します。

```
Switch(config)# line vty 10
Switch(config-line)# password let45me67in89
```

## ユーザ名とパスワードのペアの設定

ユーザ名とパスワードのペアを設定でき、スイッチ上でローカルに保存します。このペアは回線またはポートに割り当てられ、各ユーザを認証します。ユーザは認証後、スイッチにアクセスできます。権限レベルを定義している場合は、ユーザ名とパスワードの各ペアに特定の権限レベル（対応する権利および権限付き）を割り当てることもできます。

ユーザ名ベースの認証システムを設定するには、イネーブル EXEC モードで次の手順を実行します。この認証システムは、ログインユーザ名とパスワードを要求します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>username name [privilege level] {password encryption-type password}</code>	各ユーザのユーザ名、権限レベル、パスワードを入力します。 <ul style="list-style-type: none"> <li><code>name</code> には、ユーザ ID を 1 ワードで指定します。スペースや引用符は使用できません。</li> <li>(任意) <code>level</code> には、アクセス後ユーザに設定する権限レベルを指定します。指定できる範囲は 0 ~ 15 です。レベル 15 ではイネーブル EXEC モードでのアクセスが可能です。レベル 1 では、ユーザ EXEC モードでのアクセスとなります。</li> <li><code>encryption-type</code> には、暗号化されていないパスワードがあとに続く場合は 0 を、暗号化されたパスワードがあとに続く場合は 7 を指定します。</li> <li><code>password</code> には、ユーザがスイッチにアクセスする場合に入力する必要のあるパスワードを指定します。パスワードは 1 ~ 25 文字で、埋め込みスペースを使用でき、<code>username</code> コマンドの最後のオプションとして指定します。</li> </ul>
ステップ 3	<code>line console 0</code> または <code>line vty 0 15</code>	ライン コンフィギュレーション モードを開始し、コンソール ポート（回線 0）または VTY 回線（回線 0 ~ 15）を設定します。
ステップ 4	<code>login local</code>	ログイン時のローカル パスワード チェックをイネーブルにします。認証は、ステップ 2 で指定されたユーザ名に基づきます。
ステップ 5	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 6	<code>show running-config</code>	設定を確認します。
ステップ 7	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

特定ユーザのユーザ名認証をディセーブルにするには、`no username name` グローバル コンフィギュレーション コマンドを使用します。パスワード チェックをディセーブルにし、パスワードを指定しない接続を可能にするには、`no login` ライン コンフィギュレーション コマンドを使用します。

## 複数の権限レベルの設定

ソフトウェアはデフォルトで、ユーザ EXEC およびイネーブル EXEC という 2 種類のパスワードセキュリティ モードを備えています。各モードについて、コマンドの階層レベルを最大 16 まで設定できます。複数のパスワードを設定することにより、さまざまなユーザ グループに対して特定のコマンドへのアクセスを許可できます。

たとえば、多くのユーザに `clear line` コマンドへのアクセスを許可する場合、レベル 2 のセキュリティを割り当て、レベル 2 のパスワードを広範囲のユーザに配布できます。また、`configure` コマンドへのアクセスをより制限されたものにしたい場合は、レベル 3 のセキュリティを割り当て、そのパスワードを限られたユーザ グループに配布することもできます。



ここでは、次の設定について説明します。

- コマンドの権限レベルの設定 (p.8-9)
- 回線に対するデフォルトの権限レベルの変更 (p.8-10)
- 権限レベルへのログインおよび終了 (p.8-10)

## コマンドの権限レベルの設定

コマンドモードの権限レベルを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>privilege mode level level command</code>	コマンドの権限レベルを設定します。 <ul style="list-style-type: none"> <li>• <i>mode</i> には、グローバル コンフィギュレーション モードの場合は <b>configure</b> を、EXEC モードの場合は <b>exec</b> を、インターフェイス コンフィギュレーション モードの場合は <b>interface</b> を、ライン コンフィギュレーション モードの場合は <b>line</b> を、それぞれ入力します。</li> <li>• <i>level</i> に指定できる範囲は 0 ~ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。レベル 15 は、イネーブル パスワードによって許可されるアクセス レベルです。</li> <li>• <i>command</i> には、アクセスを制限したいコマンドを指定します。</li> </ul>
ステップ 3	<code>enable password level level password</code>	権限レベルのイネーブル パスワードを指定します。 <ul style="list-style-type: none"> <li>• <i>level</i> に指定できる範囲は 0 ~ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。</li> <li>• <i>password</i> には、1 ~ 25 文字の英数字のストリングを指定します。ストリングは数字で始めることはできません。大文字 / 小文字を区別し、スペースを使用できますが、先行スペースは無視されます。デフォルトでは、パスワードは定義されていません。</li> </ul>
ステップ 4	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 5	<code>show running-config</code>  または <code>show privilege</code>	設定を確認します。  <code>show running-config</code> コマンドは、パスワードとアクセス レベルの設定を表示します。 <code>show privilege</code> コマンドは、権限レベルの設定を表示します。
ステップ 6	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

コマンドをある権限レベルに設定すると、構文がそのコマンドのサブセットであるコマンドはすべて、そのレベルに設定されます。たとえば、**show ip traffic** コマンドをレベル 15 に設定すると、**show** コマンドおよび **show ip** コマンドは、個々に別のレベルに設定しないかぎり、自動的にレベル 15 に設定されます。

特定のコマンドについて、デフォルトの権限に戻すには、**no privilege mode level level command** グローバル コンフィギュレーション コマンドを使用します。

**configure** コマンドを権限レベル 14 に設定し、レベル 14 コマンドを使用する場合にユーザが入力するパスワードとして *SecretPswd14* を定義する例を示します。

```
Switch(config)# privilege exec level 14 configure
Switch(config)# enable password level 14 SecretPswd14
```

## 回線に対するデフォルトの権限レベルの変更

回線に対するデフォルトの権限レベルを変更するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>line vty line</code>	アクセスを制限する仮想端末回線を選択します。
ステップ 3	<code>privilege level level</code>	回線のデフォルト権限レベルを変更します。  <i>level</i> に指定できる範囲は 0 ~ 15 です。レベル 1 が通常のユーザ EXEC モード権限です。レベル 15 は、イネーブル パスワードによって許可されるアクセス レベルです。
ステップ 4	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 5	<code>show running-config</code>  または <code>show privilege</code>	設定を確認します。  <code>show running-config</code> コマンドは、パスワードとアクセス レベルの設定を表示します。 <code>show privilege</code> コマンドは、権限レベルの設定を表示します。
ステップ 6	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

ユーザは、回線にログインし、別の権限レベルをイネーブルに設定することにより、**privilege level** ライン コンフィギュレーション コマンドを使用して設定された権限レベルを上書きできます。また、**disable** コマンドを使用すれば、権限レベルを低く設定できます。上位の権限レベルのパスワードがわかっている場合、ユーザはそのパスワードを使用して上位の権限レベルをイネーブルにできます。回線の使用を制限するには、コンソール回線に高いレベルまたは権限レベルを指定してください。

回線の権限レベルをデフォルトに戻すには、**no privilege level** ライン コンフィギュレーション コマンドを使用します。

## 権限レベルへのログインおよび終了

特定の権限レベルにログインし、特定の権限レベルを終了するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>enable level</code>	特定の権限レベルにログインします。  <i>level</i> に指定できる範囲は 0 ~ 15 です。
ステップ 2	<code>disable level</code>	特定の権限レベルを終了します。  <i>level</i> に指定できる範囲は 0 ~ 15 です。

## TACACS+ によるスイッチ アクセスの制御

ここでは、TACACS+ をイネーブルにして設定する方法について説明します。TACACS+ は、詳細なアカウント情報収集、認証および許可プロセスに対して柔軟な管理を行います。TACACS+ は Authentication, Authorization, Accounting (AAA; 認証、許可、アカウント) を介して機能し、AAA コマンドを使用してのみイネーブルにできます。



(注)

ここで説明するコマンドの構文および使用方法の詳細については、『Cisco IOS Security Command Reference』 Release 12.2 を参照してください。

ここでは、次の設定情報について説明します。

- [TACACS+ の概要 \(p.8-11\)](#)
- [TACACS+ の動作 \(p.8-13\)](#)
- [TACACS+ の設定 \(p.8-13\)](#)
- [TACACS+ 設定の表示 \(p.8-18\)](#)

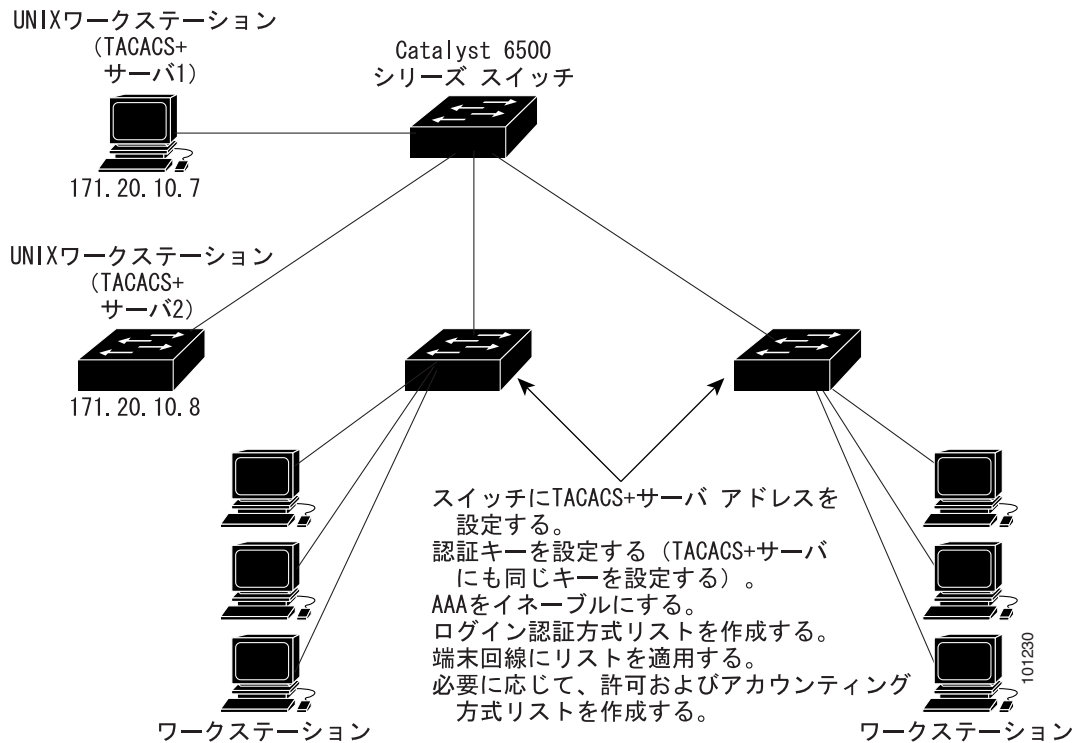
### TACACS+ の概要

TACACS+ は、スイッチにアクセスしようとするユーザの検証を集中的に行うセキュリティアプリケーションです。TACACS+ サービスは、通常 UNIX または Windows NT ワークステーション上で稼働する TACACS+ デーモンのデータベースで管理されます。スイッチに TACACS+ 機能を設定するには、TACACS+ サーバにアクセスして設定する必要があります。

TACACS+ は、個別のモジュール型認証、許可、およびアカウント機能を備えています。TACACS+ では、単一のアクセス制御サーバ (TACACS+ デーモン) が各サービス (AAA) を別個に提供します。各サービスを固有のデータベースに結合し、デーモンの機能に応じてそのサーバまたはネットワークで利用できる他のサービスを利用できます。

TACACS+ の目的は、1 つの管理サービスから複数のネットワーク アクセス ポイントを管理する方法を提供することです。スイッチは、他のシスコのルータやアクセス サーバとともにネットワーク アクセス サーバにできます。ネットワーク アクセス サーバは、単一のユーザ、ネットワークまたはサブネットワーク、および相互接続されたネットワークとの接続を実現します (図 8-1 を参照)。

図 8-1 一般的な TACACS+ ネットワーク構成



TACACS+ は、AAA セキュリティ サービスによって管理され、次のようなサービスを提供します。

- 認証 — ログインおよびパスワード ダイアログ、チャレンジおよび応答、メッセージ サポートによって認証の総合的な制御を行います。  
認証機能は、ユーザとの対話を実行できます (たとえば、ユーザ名とパスワードが入力されたあと、自宅の住所、母親の旧姓、サービス タイプ、社会保険番号などのいくつかの質問をすることによりユーザを試す)。TACACS+ 認証サービスは、ユーザ画面にメッセージを送信することもできます。たとえば、会社のパスワード有効期間ポリシーに従い、パスワードの変更の必要があることをユーザに通知することもできます。
- 許可 — オートコマンド、アクセス制御、セッション期間、プロトコル サポートの設定といった、ユーザセッション内でのユーザ機能についてきめ細かい制御を行います。また、TACACS+ 許可機能によって、ユーザが実行できるコマンドを制限することもできます。
- アカウンティング — 課金、監査、レポートに使用する情報を収集して TACACS+ デーモンに送信します。ネットワークの管理者は、アカウンティング機能を使用して、セキュリティ監査のためにユーザの活動状況を追跡したり、ユーザ課金用の情報を提供したりできます。アカウンティング レコードには、ユーザ ID、開始時刻および終了時刻、実行されたコマンド (PPP など)、パケット数、およびバイト数が含まれます。

TACACS+ プロトコルは、スイッチと TACACS+ デーモンとの間の認証を行い、スイッチと TACACS+ デーモンとの間のプロトコル交換をすべて暗号化することにより機密保持を実現します。

スイッチで TACACS+ を使用するには、TACACS+ デーモン ソフトウェアが稼働するシステムが必要です。

## TACACS+ の動作

ユーザが、TACACS+ を使用しているスイッチに対して簡易 ASCII ログインを試行し、認証が必要になると、次のプロセスが発生します。

1. 接続が確立すると、スイッチは TACACS+ デーモンに接続してユーザ名プロンプトを取得し、これがユーザに表示されます。ユーザがユーザ名を入力すると、スイッチは TACACS+ デーモンに接続してパスワードプロンプトを取得します。スイッチがパスワードプロンプトを表示し、ユーザがパスワードを入力すると、そのパスワードが TACACS+ デーモンに送信されます。  
TACACS+ によって、デーモンとユーザとの間の会話が可能になり、デーモンはユーザを認証できるだけの情報を取得できるようになります。デーモンは、ユーザ名とパスワードの組み合わせを入力するよう指示しますが、ユーザの母親の旧姓など、その他の項目を含めることもできます。
2. スイッチは、最終的に TACACS+ デーモンから次のいずれかの応答を受信します。
  - ACCEPT — ユーザが認証され、サービスを開始できます。許可を必要とするようにスイッチが設定されている場合は、この時点で許可処理が開始されます。
  - REJECT — ユーザは認証されません。TACACS+ デーモンに応じて、ユーザはアクセスを拒否されるか、ログインシーケンスを再試行するよう求められます。
  - ERROR — デーモンによる認証サービスのある時点で、またはデーモンとスイッチの間のネットワーク接続においてエラーが発生しました。ERROR 応答が表示された場合は、スイッチは、通常別の方法でユーザを認証しようとします。
  - CONTINUE — ユーザは、さらに認証情報の入力を求められます。

認証後、スイッチで許可がイネーブルになっている場合、ユーザは追加の許可フェーズに入ります。ユーザはまず、TACACS+ 認証を正常に終了しなければ TACACS+ 許可に進めません。

3. TACACS+ 許可が必要な場合は、再度 TACACS+ デーモンに接続し、デーモンが ACCEPT または REJECT の許可応答を返します。ACCEPT 応答が返された場合は、その応答には、そのユーザの EXEC または NETWORK セッション宛ての属性の形式でデータが含まれており、ユーザがアクセスできるサービスが決まります。
  - Telnet、Secure Shell (SSH; セキュア シェル)、rlogin、またはイネーブル EXEC サービス
  - 接続パラメータ (ホストまたはクライアントの IP アドレス、アクセスリスト、ユーザタイムアウトなど)

## TACACS+ の設定

ここでは、TACACS+ をサポートするようにスイッチを設定する方法について説明します。少なくとも、TACACS+ デーモンを保持するホスト (複数可) を特定し、TACACS+ 認証の方式リストを定義する必要があります。任意で TACACS+ 許可およびアカウントングの方式リストを定義することもできます。方式リストは、ユーザに対して認証、許可するか、またはアカウントを維持するための順序と方式を定義します。方式リストを使用すると、使用するセキュリティプロトコルを 1 つまたは複数指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザに対して認証、許可するか、またはアカウントの維持を行います。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の方式による通信が成功するか、方式リストを使い果たすまで続きます。

ここでは、次の設定情報について説明します。

- [TACACS+ のデフォルト設定 \(p.8-14\)](#)
- [TACACS+ サーバホストの識別と認証鍵の設定 \(p.8-14\)](#)
- [TACACS+ ログイン認証の設定 \(p.8-15\)](#)
- [イネーブル EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可の設定 \(p.8-17\)](#)
- [TACACS+ アカウントングの開始 \(p.8-18\)](#)

## TACACS+ のデフォルト設定

TACACS+ と AAA は、デフォルトでディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して TACACS+ を設定することはできません。TACACS+ をイネーブルに設定すると、CLI を使用してスイッチにアクセスするユーザを認証します。



(注) TACACS+ の設定は CLI を使用して行いますが、TACACS+ サーバは権限レベル 15 に設定された HTTP 接続を許可します。

## TACACS+ サーバ ホストの識別と認証鍵の設定

認証用に 1 つのサーバを使用することも、また、既存のサーバ ホストをグループ化するために AAA サーバ グループを使用するよう設定することもできます。設定済みサーバ ホストのサブセットを選択してサーバをグループ化し、特定のサービスに使用できます。サーバグループは、グローバルサーバホストリストとともに使用され、選択されたサーバホストの IP アドレスのリストを含んでいます。

IP ホストまたは TACACS+ サーバを保持するホストを識別し、任意で暗号鍵を設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>tacacs-server host hostname [port integer] [timeout integer] [key string]</code>	TACACS+ サーバを保持する IP ホストまたはホストを識別します。このコマンドを複数回入力すると、優先ホストのリストが作成されます。ソフトウェアは、指定された順序でホストを検索します。 <ul style="list-style-type: none"> <li><code>hostname</code> には、ホストの名前または IP アドレスを指定します。</li> <li>(任意) <code>port integer</code> には、サーバのポート番号を指定します。デフォルトはポート 49 です。指定できる範囲は 1 ~ 65535 です。</li> <li>(任意) <code>timeout integer</code> には、スイッチがデーモンからの応答を待つ時間を秒数で指定します。これを過ぎるとスイッチは時間切れとなりエラーを宣言します。デフォルト値は 5 秒です。指定できる範囲は 1 ~ 1000 秒です。</li> <li>(任意) <code>key string</code> には、スイッチと TACACS+ デーモンとの間のすべてのトラフィックを暗号化および暗号解除するための暗号鍵を指定します。暗号化が成功するには TACACS+ デーモンに同じ鍵を設定する必要があります。</li> </ul>
ステップ 3	<code>aaa new-model</code>	AAA をイネーブルにします。
ステップ 4	<code>aaa group server tacacs+ group-name</code>	(任意) グループ名で AAA サーバグループを定義します。 このコマンドによって、スイッチはサーバ グループ サブコンフィギュレーション モードになります。
ステップ 5	<code>server ip-address</code>	(任意) 特定の TACACS+ サーバを定義済みサーバグループに対応付けます。AAA サーバグループの各 TACACS+ サーバに対してこのステップを繰り返します。 グループの各サーバは、ステップ 2 で定義済みのものでなければなりません。
ステップ 6	<code>end</code>	イネーブル EXEC モードに戻ります。

	コマンド	説明
ステップ 7	<b>show tacacs</b>	設定を確認します。
ステップ 8	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

指定された TACACS+ サーバ名またはアドレスを削除するには、**no tacacs-server host hostname** グローバル コンフィギュレーション コマンドを使用します。設定リストからサーバ グループを削除するには、**no aaa group server tacacs+ group-name** グローバル コンフィギュレーション コマンドを使用します。TACACS+ サーバの IP アドレスを削除するには、**no server ip-address** サーバグループ サブコンフィギュレーション コマンドを使用します。

## TACACS+ ログイン認証の設定

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、さまざまなポートにそのリストを適用します。方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外はデフォルトの方式リスト（偶然に *default* と名前が付けられている）です。デフォルトの方式リストは、名前付き方式リストが明示的に定義されたポートを除いて、自動的にすべてのインターフェイスに適用されます。定義済みの方式リストは、デフォルトの方式リストに優先します。

方式リストは、ユーザ認証のためクエリ送信を行う順番と認証方式を記述したものです。認証に使用する 1 つまたは複数のセキュリティ プロトコルを指定できるので、最初の方式が失敗した場合のバックアップ システムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試行するまで続きます。この処理のある時点で認証が失敗した場合（つまり、セキュリティ サーバまたはローカルのユーザ名データベースがユーザ アクセスを拒否すると応答した場合）、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

ログイン認証を設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>aaa new-model</b>	AAA をイネーブルにします。

	コマンド	説明
ステップ 3	<code>aaa authentication login {default   list-name} method1 [method2...]</code>	<p>ログイン認証方式リストを作成します。</p> <ul style="list-style-type: none"> <li>• <b>login authentication</b> コマンドに名前付きリストが指定されていない場合に使用されるデフォルトのリストを作成するには、<b>default</b> キーワードの後ろにデフォルト状況で使用する方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。</li> <li>• <i>list-name</i> には、作成するリストの名前として使用する文字列を指定します。</li> <li>• <i>method1...</i> には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。</li> </ul> <p>次のいずれかの方式を選択します。</p> <ul style="list-style-type: none"> <li>• <b>enable</b> — イネーブル パスワードを認証に使用します。この認証方式を使用するには、<b>enable password</b> グローバル コンフィギュレーション コマンドを使用して、イネーブル パスワードをあらかじめ定義しておく必要があります。</li> <li>• <b>group tacacs+</b> — TACACS+ 認証を使用します。この認証方式を使用するには、TACACS+ サーバをあらかじめ設定しておく必要があります。詳細については、「<a href="#">TACACS+ サーバ ホストの識別と認証鍵の設定</a>」(p.8-14) を参照してください。</li> <li>• <b>line</b> — 回線パスワードを認証に使用します。この認証方式を使用するには、回線パスワードをあらかじめ設定しておく必要があります。<b>password password</b> ライン コンフィギュレーション コマンドを使用します。</li> <li>• <b>local</b> — ローカル ユーザ名データベースを認証に使用します。データベースにユーザ名情報を入力しておく必要があります。<b>username password</b> グローバル コンフィギュレーション コマンドを使用します。</li> <li>• <b>local-case</b> — 大文字 / 小文字を区別するローカル ユーザ名データベースを認証に使用します。<b>username name password</b> グローバル コンフィギュレーション コマンドを使用して、データベースにユーザ名情報を入力する必要があります。</li> <li>• <b>none</b> — ログインに認証を使用しません。</li> </ul>
ステップ 4	<code>line [console   tty   vty] line-number [ending-line-number]</code>	ライン コンフィギュレーション モードを開始し、認証リストの適用対象とする回線を設定します。
ステップ 5	<code>login authentication {default   list-name}</code>	<p>回線または回線セットに対して、認証リストを適用します。</p> <ul style="list-style-type: none"> <li>• <b>default</b> を指定する場合は、<b>aaa authentication login</b> コマンドで作成したデフォルトのリストを使用します。</li> <li>• <i>list-name</i> には、<b>aaa authentication login</b> コマンドで作成したリストを指定します。</li> </ul>
ステップ 6	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 7	<code>show running-config</code>	設定を確認します。
ステップ 8	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。



AAA をディセーブルにするには、**no aaa new-model** グローバル コンフィギュレーション コマンドを使用します。AAA 認証をディセーブルにするには、**no aaa authentication login {default | list-name} method1 [method2...]** グローバル コンフィギュレーション コマンドを使用します。ログインの TACACS+ 認証をディセーブルにするかデフォルト値に戻す場合は、**no login authentication {default | list-name}** ライン コンフィギュレーション コマンドを使用します。

## イネーブル EXEC アクセスおよびネットワーク サービス用の TACACS+ 許可の設定

AAA 許可は、ユーザが利用できるサービスを制限します。AAA 許可がイネーブルに設定されていると、スイッチはユーザのプロファイルから取得した情報を使用します。このプロファイルは、ローカルのユーザ データベースまたはセキュリティ サーバ上にあり、ユーザのセッションを設定します。ユーザは、ユーザ プロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

**aaa authorization** グローバル コンフィギュレーション コマンドに **tacacs+** キーワードを付けて使用すると、イネーブル EXEC モードへのユーザのネットワーク アクセスを制限するパラメータを設定できます。

**aaa authorization exec tacacs+ local** コマンドは、次の許可パラメータを設定します。

- TACACS+ を使用して認証を行った場合は、イネーブル EXEC アクセス許可に TACACS+ を使用します。
- 認証に TACACS+ を使用しなかった場合は、ローカル データベースを使用します。



(注) 許可が設定されていても、CLI 経由でログインして認証されたユーザに対して、許可は省略されます。

イネーブル EXEC アクセスおよびネットワーク サービスに関する TACACS+ 許可を指定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>aaa authorization network tacacs+</b>	ネットワーク関連のすべてのサービス要求に対するユーザ TACACS+ 許可をスイッチに設定します。
ステップ 3	<b>aaa authorization exec tacacs+</b>	イネーブル EXEC アクセスの有無を、ユーザ TACACS+ 許可によって判別するようにスイッチを設定します。  <b>exec</b> キーワードを指定すると、ユーザプロファイル情報 ( <b>autocommand</b> 情報など) を返すことができます。
ステップ 4	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 5	<b>show running-config</b>	設定を確認します。
ステップ 6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

許可をディセーブルにするには、**no aaa authorization {network | exec} method1** グローバル コンフィギュレーション コマンドを使用します。

## TACACS+ アカウンティングの開始

AAA アカウンティング機能は、ユーザがアクセスするサービスと、ユーザが消費するネットワークリソースを追跡します。AAA アカウンティングがイネーブルに設定されていると、スイッチは、アカウンティングレコードの形式でユーザの活動状況を TACACS+ セキュリティ サーバに報告します。各アカウンティングレコードには、アカウンティングの Attribute-Value (AV) のペアが含まれ、セキュリティサーバ上に保存されます。このデータを分析し、ネットワーク管理、クライアントへの課金、または監査に利用できます。

各 Cisco IOS 権限レベルおよびネットワークサービスに関する TACACS+ アカウンティングをイネーブルにするには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>aaa accounting network start-stop tacacs+</code>	ネットワーク関連のすべてのサービス要求に対する TACACS+ アカウンティングをイネーブルにします。
ステップ 3	<code>aaa accounting exec start-stop tacacs+</code>	TACACS+ アカウンティングにより、イネーブル EXEC プロセスの開始時に記録開始アカウンティング通知、終了時に記録停止通知を送信するように設定します。
ステップ 4	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 5	<code>show running-config</code>	設定を確認します。
ステップ 6	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

アカウンティングをディセーブルにするには、`no aaa accounting {network | exec} {start-stop} method1...` グローバル コンフィギュレーション コマンドを使用します。

## TACACS+ 設定の表示

TACACS+ サーバ統計情報を表示するには、`show tacacs` イネーブル EXEC コマンドを使用します。

## RADIUS によるスイッチ アクセスの制御

ここでは、RADIUS をイネーブルにして設定する方法について説明します。RADIUS は、詳細なアカウント情報収集、認証および許可プロセスに対して柔軟な管理を行います。RADIUS は AAA を介して機能し、AAA コマンドを使用するのみイネーブルにできます。



(注) ここで説明するコマンドの構文および使用方法の詳細については、『Cisco IOS Security Command Reference』 Release 12.2 を参照してください。

ここでは、次の設定情報について説明します。

- RADIUS の概要 (p.8-19)
- RADIUS の動作 (p.8-20)
- RADIUS の設定 (p.8-21)
- RADIUS 設定の表示 (p.8-32)

### RADIUS の概要

RADIUS は分散型クライアント/サーバシステムで、不正なアクセスからネットワークを保護します。RADIUS クライアントは、サポートされているシスコのルータおよびスイッチ上で稼働し、中央 RADIUS サーバに認証要求を送信します。中央 RADIUS サーバには、すべてのユーザの認証およびネットワーク サービス アクセス情報が格納されています。RADIUS ホストは、通常、シスコ (Cisco Secure Access Control Server バージョン 3.0)、Livingston、Merit、Microsoft などのソフトウェア製造元の RADIUS サーバソフトウェアが稼働するマルチユーザシステムです。詳細については、RADIUS サーバのマニュアルを参照してください。

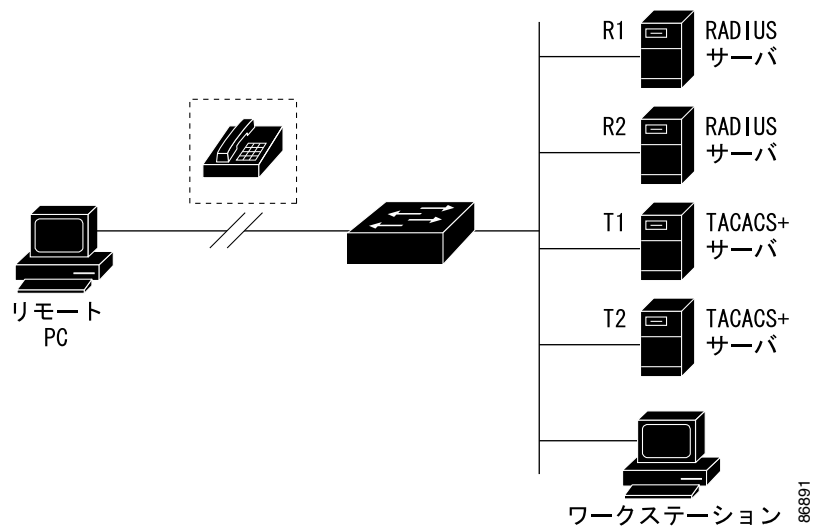
RADIUS は、アクセスのセキュリティが必要な、次のネットワーク環境で使用します。

- それぞれが RADIUS をサポートする、マルチベンダー アクセス サーバによるネットワーク。たとえば、複数のベンダーのアクセス サーバが、1 つの RADIUS サーバベース セキュリティ データベースを使用します。複数のベンダーのアクセス サーバからなる IP ベースのネットワークでは、ダイヤルイン ユーザは RADIUS サーバを通じて認証されます。RADIUS サーバは、Kerberos セキュリティ システムで動作するようにカスタマイズされています。
- アプリケーションが RADIUS プロトコルをサポートするターンキー ネットワーク セキュリティ環境。たとえば、スマート カードアクセス制御システムを使用するアクセス環境。あるケースでは、RADIUS は Enigma のセキュリティ カードと併用してユーザを確認し、ネットワーク リソースのアクセスを許可してきました。
- すでに RADIUS を使用中のネットワーク。RADIUS クライアントを含むシスコのスイッチをネットワークに追加できます。これが TACACS+ サーバへの移行の最初のステップとなることもあります (図 8-2[p.8-20] を参照)。
- ユーザが 1 つのサービスにしかアクセスできないネットワーク。RADIUS を使用すると、ユーザのアクセスを 1 つのホスト、Telnet などの 1 つのユーティリティ、または IEEE 802.1x などのプロトコルを使用するネットワークに制御できます。このプロトコルの詳細については、第 9 章「802.1x ポートベースの認証の設定」を参照してください。
- リソース アカウンティングが必要なネットワーク。RADIUS 認証または許可とは別に RADIUS アカウンティングを使用できます。RADIUS アカウンティング機能によって、サービスの開始および終了時点でデータを送信し、このセッション中に使用されるリソース (時間、パケット、バイトなど) の量を表示できます。インターネット サービス プロバイダーは、RADIUS アクセス制御およびアカウンティング ソフトウェアのフリーウェア バージョンを使用して、特殊なセキュリティおよび請求のニーズを満たすこともできます。

RADIUS は、ネットワーク セキュリティが次のような状況には適していません。

- マルチプロトコルアクセス環境。RADIUS は、AppleTalk Remote Access (ARA)、NetBIOS Frame Control Protocol (NBFCP)、NetWare Asynchronous Services Interface (NASI)、または X.25 PAD 接続をサポートしません。
- スイッチ間またはルータ間。RADIUS は、双方向認証を行いません。RADIUS は、他社製のデバイスが認証を必要とする場合に、あるデバイスから他社製デバイスへの認証には使用できません。
- 各種のサービスを使用するネットワーク。RADIUS は、一般に 1 人のユーザを 1 つのサービスモデルにバインドします。

図 8-2 RADIUS から TACACS+ サービスへの移行



## RADIUS の動作

RADIUS サーバによってアクセス制御されているスイッチに、ユーザがログインして認証を試行すると、次のイベントが発生します。

1. ユーザ名とパスワードの入力を求めるプロンプトが表示されます。
2. ユーザ名と暗号化されたパスワードが、ネットワークを介して RADIUS サーバに送信されます。
3. ユーザは、RADIUS サーバから次のいずれかの応答を受信します。
  - a. ACCEPT — ユーザは認証されます。
  - b. REJECT — ユーザは認証されず、ユーザ名とパスワードの再入力を求めるプロンプトが表示されるか、アクセスが拒否されます。
  - c. CHALLENGE — ユーザからの追加データが要求されます。
  - d. CHALLENGE PASSWORD — ユーザは新しいパスワードを選択するよう要求されます。

ACCEPT または REJECT 応答には、イネーブル EXEC またはネットワーク許可に使用される追加データがバンドルされています。RADIUS 許可がイネーブルになっている場合、ユーザはまず、RADIUS 認証に成功しなければ RADIUS 許可に進めません。ACCEPT または REJECT パケットの追加データには、次の項目が含まれています。

- Telnet、SSH、rlogin、またはイネーブル EXEC サービス
- 接続パラメータ (ホストまたはクライアントの IP アドレス、アクセス リスト、ユーザ タイムアウトなど)

## RADIUS の設定

ここでは、RADIUS をサポートするようにスイッチを設定する方法について説明します。少なくとも、RADIUS サーバソフトウェアを稼働するホスト（複数可）を特定し、RADIUS 認証の方式リストを定義する必要があります。任意で RADIUS 許可およびアカウントिंगの方式リストを定義することもできます。

方式リストは、ユーザに対して認証、許可するか、またはアカウントを維持するための順序と方式を定義します。方式リストを使用すると、使用するセキュリティプロトコル（TACACS+ やローカルユーザ名検索など）を 1 つまたは複数指定できるので、最初の方式が失敗した場合のバックアップシステムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザに対して認証、許可するか、またはアカウントの維持を行います。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の方式による通信が成功するか、方式リストを使い果たすまで続きます。

スイッチに RADIUS 機能を設定するには、RADIUS サーバにアクセスして設定する必要があります。ここでは、次の設定情報について説明します。

- [RADIUS のデフォルト設定 \(p.8-21\)](#)
- [RADIUS サーバホストの識別 \(p.8-21\)](#) (必須)
- [RADIUS ログイン認証の設定 \(p.8-24\)](#) (必須)
- [AAA サーバグループの定義 \(p.8-26\)](#) (任意)
- [ユーザイネーブルアクセスおよびネットワーク サービス用の RADIUS 許可の設定 \(p.8-28\)](#) (任意)
- [RADIUS アカウントिंगの開始 \(p.8-29\)](#) (任意)
- [すべての RADIUS サーバに対する設定 \(p.8-30\)](#) (任意)
- [ベンダー固有の RADIUS アトリビュート用にスイッチを設定する方法 \(p.8-30\)](#) (任意)
- [ベンダー固有の RADIUS サーバ通信用にスイッチを設定する方法 \(p.8-32\)](#) (任意)

## RADIUS のデフォルト設定

RADIUS と AAA は、デフォルトでディセーブルに設定されています。

セキュリティの失効を防止するため、ネットワーク管理アプリケーションを使用して RADIUS を設定することはできません。RADIUS をイネーブルに設定すると、CLI を使用して、スイッチにアクセスするユーザを認証します。

## RADIUS サーバホストの識別

スイッチと RADIUS サーバ間の通信には、次の要素が関係します。

- ホスト名または IP アドレス
- 認証宛先ポート
- アカウントिंग宛先ポート
- キー strings
- タイムアウト時間
- 再送信回数

RADIUS セキュリティ サーバは、ホスト名または IP アドレス、ホスト名と各 UDP ポート番号、あるいは IP アドレスと各 UDP ポート番号で識別されます。IP アドレスと UDP ポート番号の組み合わせによって一意の識別子が作成され、特定の AAA サービスを提供する RADIUS ホストとしてさまざまなポートを個別に定義できます。この一意の識別子によって、サーバ上の複数の UDP ポートに同じ IP アドレスで RADIUS 要求を送信できるようになります。

同一の RADIUS サーバ上の 2 つの異なるホスト エントリが同じサービス (たとえば、アカウントティング) を設定している場合、設定された 2 番めのホスト エントリは、最初のエントリの代替バックアップとして機能します。この例では、最初のホスト エントリがアカウントティング サービスを提供できない場合は、スイッチは、同じデバイス上に設定された 2 番めのホスト エントリでアカウントティング サービスを試行します (RADIUS のホスト エントリは、設定された順序で試行されます)。

RADIUS サーバおよびスイッチは、共有シークレット テキスト ストリングを使用してパスワードを暗号化し、応答を交換します。AAA セキュリティ コマンドを使用するように RADIUS を設定するには、RADIUS サーバ デーモンが稼働するホスト、およびスイッチと共有するシークレット テキスト (キー) ストリングを指定する必要があります。

タイムアウト、再送信回数、および暗号化鍵の値は、すべての RADIUS サーバに対してグローバルにサーバ単位で設定することも、グローバルな設定とサーバ単位の設定を組み合わせることもできます。この設定を、スイッチと通信するすべての RADIUS サーバに対してグローバルに適用するには、3 つの特別なグローバル コンフィギュレーション コマンド、**radius-server timeout**、**radius-server retransmit**、および **radius-server key** を使用します。特定の RADIUS サーバにこれらの値を適用するには、**radius-server host** グローバル コンフィギュレーション コマンドを使用します。




(注)

スイッチにグローバルおよびサーバ単位の両方で機能 (タイムアウト、再送信回数、およびキー コマンド) を設定すると、サーバ単位のタイマー、再送信回数、およびキー コマンドは、グローバルのタイマー、再送信、およびキー コマンドを上書きします。すべての RADIUS サーバに対してこれらの値を設定するには、「[すべての RADIUS サーバに対する設定](#)」(p.8-30) を参照してください。

認証用に既存のサーバ ホストをグループ化するために、AAA サーバ グループを使用するようスイッチを設定できます。詳細については、「[AAA サーバ グループの定義](#)」(p.8-26) を参照してください。

サーバ単位での RADIUS サーバ通信を設定するには、イネーブル EXEC モードで次の手順を実行します。この手順は必須です。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>radius-server host {hostname   ip-address} [auth-port port-number] [acct-port port-number] [timeout seconds] [retransmit retries] [key string]</code>	<p>リモート RADIUS サーバホストの IP アドレスまたはホスト名を指定します。</p> <ul style="list-style-type: none"> <li>• (任意) <code>auth-port port-number</code> には、認証要求の UDP 宛先ポートを指定します。</li> <li>• (任意) <code>acct-port port-number</code> には、アカウント要求の UDP 宛先ポートを指定します。</li> <li>• (任意) <code>timeout seconds</code> には、スイッチが再送信するまでに RADIUS サーバの応答を待機するインターバルを指定します。指定できる範囲は 1 ~ 1000 です。この設定は、<code>radius-server timeout</code> グローバル コンフィギュレーション コマンドの設定を上書きします。<code>radius-server host</code> コマンドでタイムアウトを設定しない場合は、<code>radius-server timeout</code> コマンドの設定が使用されます。</li> <li>• (任意) <code>retransmit retries</code> には、サーバが応答しないか、応答が遅い場合に、RADIUS 要求をそのサーバに再送信する回数を指定します。指定できる範囲は 1 ~ 1000 です。<code>radius-server host</code> コマンドで再送信の値を設定しない場合は、<code>radius-server retransmit</code> グローバル コンフィギュレーション コマンドの設定が使用されます。</li> <li>• (任意) <code>key string</code> には、スイッチと RADIUS サーバ上で稼働する RADIUS デーモンとの間で使用する認証および暗号化鍵を指定します。</li> </ul> <p> (注) 鍵は、RADIUS サーバ上で使用する暗号化鍵と照合する必要があるテキストストリングです。鍵は、必ず <code>radius-server host</code> コマンドの最後の項目として設定します。先行スペースは無視されますが、鍵の途中および末尾のスペースは使用されます。鍵にスペースを使用する場合は、鍵の一部として引用符を使用する場合を除いて、鍵を引用符で囲まないでください。</p> <p>1 つの IP アドレスに対応する複数のホスト エントリをスイッチが認識するように設定するには、必要な回数だけこのコマンドを入力し、それぞれの UDP ポート番号が必ず異なるようにしてください。スイッチ ソフトウェアは、指定された順序でホストを検索します。特定の RADIUS ホストで使用するタイムアウト、再送信回数、および暗号化鍵の値を設定します。</p>
ステップ 3	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 4	<code>show running-config</code>	設定を確認します。
ステップ 5	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

特定の RADIUS サーバを削除するには、`no radius-server host hostname | ip-address` グローバル コンフィギュレーション コマンドを使用します。

次に、ある RADIUS サーバを認証用に、別の RADIUS サーバをアカウントング用に設定する例を示します。

```
Switch(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Switch(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

次の例は、RADIUS サーバとして *host1* を設定し、認証およびアカウントングの両方にデフォルトポートを使用する方法を示します。

```
Switch(config)# radius-server host host1
```



(注) さらに、RADIUS サーバでいくつかの設定を行う必要があります。この設定とは、スイッチの IP アドレス、およびサーバとスイッチで共有するキー ストリングです。詳細については、RADIUS サーバのマニュアルを参照してください。

## RADIUS ログイン認証の設定

AAA 認証を設定するには、認証方式の名前付きリストを作成してから、さまざまなポートにそのリストを適用します。方式リストは実行される認証のタイプと実行順序を定義します。このリストを特定のポートに適用してから、定義済み認証方式を実行する必要があります。唯一の例外はデフォルトの方式リスト（偶然に *default* と名前が付けられている）です。デフォルトの方式リストは、名前付き方式リストが明示的に定義されたポートを除いて、自動的にすべてのインターフェイスに適用されます。

方式リストは、ユーザ認証のためクエリ送信を行う順番と認証方式を記述したものです。認証に使用する 1 つまたは複数のセキュリティ プロトコルを指定できるので、最初の方式が失敗した場合のバックアップ システムが確保されます。ソフトウェアは、リスト内の最初の方式を使用してユーザを認証します。その方式で応答が得られなかった場合、ソフトウェアはそのリストから次の方式を選択します。このプロセスは、リスト内の認証方式による通信が成功するか、定義された方式をすべて試行するまで続きます。この処理のある時点で認証が失敗した場合（つまり、セキュリティサーバまたはローカルのユーザ名データベースがユーザ アクセスを拒否すると応答した場合）、認証プロセスは停止し、それ以上認証方式が試行されることはありません。

ログイン認証を設定するには、イネーブル EXEC モードで次の手順を実行します。この手順は必須です。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>aaa new-model</code>	AAA をイネーブルにします。



	コマンド	説明
ステップ 3	<b>aaa authentication login</b> {default   list-name} method1 [method2...]	<p>ログイン認証方式リストを作成します。</p> <ul style="list-style-type: none"> <li>• <b>login authentication</b> コマンドに名前付きリストが指定されていない場合に使用されるデフォルトのリストを作成するには、<b>default</b> キーワードの後ろにデフォルト状況で使用する方式を指定します。デフォルトの方式リストは、自動的にすべてのポートに適用されます。</li> <li>• <i>list-name</i> には、作成するリストの名前として使用する文字列を指定します。</li> <li>• <i>method1...</i> には、認証アルゴリズムが試行する実際の方式を指定します。追加の認証方式は、その前の方式でエラーが返された場合に限り使用されます。前の方式が失敗した場合は使用されません。</li> </ul> <p>次のいずれかの方式を選択します。</p> <ul style="list-style-type: none"> <li>— <b>enable</b> — イネーブル パスワードを認証に使用します。この認証方式を使用するには、<b>enable password</b> グローバル コンフィギュレーション コマンドを使用して、イネーブル パスワードをあらかじめ定義しておく必要があります。</li> <li>— <b>group radius</b> — RADIUS 認証を使用します。この認証方式を使用するには、RADIUS サーバをあらかじめ設定しておく必要があります。詳細については、「<a href="#">RADIUS サーバホストの識別</a>」(p.8-21) を参照してください。</li> <li>— <b>line</b> — 回線パスワードを認証に使用します。この認証方式を使用するには、回線パスワードをあらかじめ設定しておく必要があります。<b>password password</b> ライン コンフィギュレーション コマンドを使用します。</li> <li>— <b>local</b> — ローカル ユーザ名データベースを認証に使用します。データベースにユーザ名情報を入力しておく必要があります。<b>username name password</b> グローバル コンフィギュレーション コマンドを使用します。</li> <li>— <b>local-case</b> — 大文字 / 小文字を区別するローカル ユーザ名データベースを認証に使用します。<b>username password</b> グローバル コンフィギュレーション コマンドを使用して、データベースにユーザ名情報を入力する必要があります。</li> <li>— <b>none</b> — ログインに認証を使用しません。</li> </ul>
ステップ 4	<b>line</b> [console   tty   vty] line-number [ending-line-number]	ライン コンフィギュレーション モードを開始し、認証リストの適用対象とする回線を設定します。
ステップ 5	<b>login authentication</b> {default   list-name}	<p>回線または回線セットに対して、認証リストを適用します。</p> <ul style="list-style-type: none"> <li>• <b>default</b> を指定する場合は、<b>aaa authentication login</b> コマンドで作成したデフォルトのリストを使用します。</li> <li>• <i>list-name</i> には、<b>aaa authentication login</b> コマンドで作成したリストを指定します。</li> </ul>
ステップ 6	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 7	<b>show running-config</b>	設定を確認します。
ステップ 8	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

AAA をディセーブルにするには、**no aaa new-model** グローバル コンフィギュレーション コマンドを使用します。AAA 認証をディセーブルにするには、**no aaa authentication login {default | list-name} method1 [method2...]** グローバル コンフィギュレーション コマンドを使用します。ログインの RADIUS 認証をディセーブルにするかデフォルト値に戻す場合は、**no login authentication {default | list-name}** ライン コンフィギュレーション コマンドを使用します。


## AAA サーバ グループの定義

認証用に既存のサーバ ホストをグループ化するために、AAA サーバ グループを使用するようスイッチを設定できます。設定済みサーバ ホストのサブセットを選択し、特定のサービスに使用できます。サーバ グループには、グローバル サーバホスト リストを使用します。このリストは、選択したサーバ ホストの IP アドレスのリストです。

サーバ グループには、各エントリが一意的識別子 (IP アドレスと UDP ポート番号の組み合わせ) を持っていれば、同じサーバに対して複数のホスト エントリを組み込むことができます。また、特定の AAA サービスを提供する RADIUS ホストとして、さまざまなポートを個別に定義できます。同一の RADIUS サーバ上の 2 つの異なるホスト エントリを同じサービス (たとえば、アカウントティング) を設定すると、設定された 2 番めのホスト エントリは、最初のエントリの代替バックアップとして機能します。

定義済みのグループサーバに特定のサーバを対応付けるには、**server** グループ サーバ コンフィギュレーション コマンドを使用します。IP アドレスでサーバを特定したり、任意の **auth-port** および **acct-port** キーワードを使用して複数のホスト インスタンスまたはエントリを識別することもできます。

AAA サーバグループを定義してそれを特定の RADIUS サーバに対応付けるには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>radius-server host {hostname   ip-address} [auth-port port-number] [acct-port port-number] [timeout seconds] [retransmit retries] [key string]</code>	<p>リモート RADIUS サーバホストの IP アドレスまたはホスト名を指定します。</p> <ul style="list-style-type: none"> <li>（任意）<code>auth-port port-number</code> には、認証要求の UDP 宛先ポートを指定します。</li> <li>（任意）<code>acct-port port-number</code> には、アカウント要求の UDP 宛先ポートを指定します。</li> <li>（任意）<code>timeout seconds</code> には、スイッチが再送信するまでに RADIUS サーバの応答を待機するインターバルを指定します。指定できる範囲は 1 ~ 1000 です。この設定は、<code>radius-server timeout</code> グローバル コンフィギュレーション コマンドの設定を上書きします。<code>radius-server host</code> コマンドでタイムアウトを設定しない場合は、<code>radius-server timeout</code> コマンドの設定が使用されます。</li> <li>（任意）<code>retransmit retries</code> には、サーバが応答しないか、応答が遅い場合に、RADIUS 要求をそのサーバに再送信する回数を指定します。指定できる範囲は 1 ~ 1000 です。<code>radius-server host</code> コマンドで再送信の値を設定しない場合は、<code>radius-server retransmit</code> グローバル コンフィギュレーション コマンドの設定が使用されます。</li> <li>（任意）<code>key string</code> には、スイッチと RADIUS サーバ上で稼働する RADIUS デーモンとの間で使用する認証および暗号化鍵を指定します。</li> </ul> <p> (注) 鍵は、RADIUS サーバ上で使用する暗号化鍵と照合する必要があるテキストストリングです。鍵は、必ず <code>radius-server host</code> コマンドの最後の項目として設定します。先行スペースは無視されますが、鍵の途中および末尾のスペースは使用されます。鍵にスペースを使用する場合は、鍵の一部として引用符を使用する場合を除いて、鍵を引用符で囲まないでください。</p> <p>1 つの IP アドレスに対応する複数のホスト エントリをスイッチが認識するように設定するには、必要な回数だけこのコマンドを入力し、それぞれの UDP ポート番号が必ず異なるようにしてください。スイッチ ソフトウェアは、指定された順序でホストを検索します。特定の RADIUS ホストで使用するタイムアウト、再送信回数、および暗号化鍵の値を設定します。</p>
ステップ 3	<code>aaa new-model</code>	AAA をイネーブルにします。
ステップ 4	<code>aaa group server radius group-name</code>	<p>グループ名で AAA サーバグループを定義します。</p> <p>このコマンドによって、スイッチはサーバグループ コンフィギュレーション モードになります。</p>
ステップ 5	<code>server ip-address</code>	<p>特定の RADIUS サーバを定義済みサーバグループに対応付けます。AAA サーバグループの RADIUS サーバごとに、このステップを繰り返します。</p> <p>グループの各サーバは、ステップ 2 で定義済みのものでなければなりません。</p>
ステップ 6	<code>end</code>	イネーブル EXEC モードに戻ります。

	コマンド	説明
ステップ 7	<code>show running-config</code>	設定を確認します。
ステップ 8	<code>copy running-config startup-config</code>	(任意) コンフィギュレーションファイルに設定を保存します。
ステップ 9		RADIUS ログイン認証をイネーブルにします。「 <a href="#">RADIUS ログイン認証の設定</a> 」(p.8-24) を参照してください。

特定の RADIUS サーバを削除するには、`no radius-server host hostname | ip-address` グローバル コンフィギュレーション コマンドを使用します。コンフィギュレーション リストからサーバグループを削除するには、`no aaa group server radius group-name` グローバル コンフィギュレーション コマンドを使用します。RADIUS サーバの IP アドレスを削除するには、`no server ip-address` サーバグループコンフィギュレーション コマンドを使用します。

次の例では、2 つの異なる RADIUS グループサーバ (`group1` と `group2`) を認識するようにスイッチを設定しています。`group1` では、同一の RADIUS サーバ上の 2 つの異なるホスト エントリに同じサービスを設定しています。2 番目のホスト エントリは、最初のエントリの代替バックアップとして機能します。

```
Switch(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Switch(config)# aaa new-model
Switch(config)# aaa group server radius group1
Switch(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Switch(config-sg-radius)# exit
Switch(config)# aaa group server radius group2
Switch(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Switch(config-sg-radius)# exit
```

## ユーザ イネーブル アクセスおよびネットワーク サービス用の RADIUS 許可の設定

AAA 許可は、ユーザが利用できるサービスを制限します。AAA 許可がイネーブルに設定されていると、スイッチはユーザのプロファイルから取得した情報を使用します。このプロファイルは、ローカルのユーザ データベースまたはセキュリティ サーバ上にあり、ユーザのセッションを設定します。ユーザは、ユーザ プロファイル内の情報で認められている場合に限り、要求したサービスのアクセスが認可されます。

`aaa authorization` グローバル コンフィギュレーション コマンドに `radius` キーワードを付けて使用すると、イネーブル EXEC モードへのユーザのネットワーク アクセスを制限するパラメータを設定できます。

`aaa authorization exec radius local` コマンドは、次の許可パラメータを設定します。

- RADIUS を使用して認証を行った場合は、イネーブル EXEC アクセス許可に RADIUS を使用します。
- 認証に RADIUS を使用しなかった場合は、ローカル データベースを使用します。



(注) 許可が設定されていても、CLI 経由でログインして認証されたユーザに対して、許可は省略されません。

イネーブル EXEC アクセスおよびネットワーク サービスに関する RADIUS 許可を指定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>aaa authorization network radius</b>	ネットワーク関連のすべてのサービス要求に対するユーザ RADIUS 許可をスイッチに設定します。
ステップ 3	<b>aaa authorization exec radius</b>	イネーブル EXEC アクセスの有無を、ユーザ RADIUS 許可によって判別するようにスイッチを設定します。  <b>exec</b> キーワードを指定すると、ユーザ プロファイル情報 ( <b>autocommand</b> 情報など) を返すことができます。
ステップ 4	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 5	<b>show running-config</b>	設定を確認します。
ステップ 6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

許可をディセーブルにするには、**no aaa authorization {network | exec} method1** グローバル コンフィギュレーション コマンドを使用します。

## RADIUS アカウンティングの開始

AAA アカウンティング機能は、ユーザがアクセスするサービスと、ユーザが消費するネットワーク リソースを追跡します。AAA アカウンティングがイネーブルに設定されていると、スイッチは、アカウンティング レコードの形式でユーザの活動状況を RADIUS セキュリティ サーバに報告します。各アカウンティング レコードには、アカウンティングの AV のペアが含まれ、セキュリティ サーバ上に保存されます。このデータを分析し、ネットワーク管理、クライアントへの課金、または監査に利用できます。


各 Cisco IOS 権限レベルおよびネットワーク サービスに関する RADIUS アカウンティングをイネーブルにするには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>aaa accounting network start-stop radius</b>	ネットワーク関連のすべてのサービス要求に関する RADIUS アカウンティングをイネーブルにします。
ステップ 3	<b>aaa accounting exec start-stop radius</b>	RADIUS アカウンティングにより、イネーブル EXEC プロセスの開始時に記録開始アカウンティング通知、終了時に記録停止通知を送信するように設定します。
ステップ 4	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 5	<b>show running-config</b>	設定を確認します。
ステップ 6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

アカウンティングをディセーブルにするには、**no aaa accounting {network | exec} {start-stop} method1...** グローバル コンフィギュレーション コマンドを使用します。

## すべての RADIUS サーバに対する設定

スイッチとすべての RADIUS サーバ間のグローバル通信設定を設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>radius-server key string</code>	スイッチとすべての RADIUS サーバとの間で使用する、共有シークレット テキスト ストリングを指定します。   (注) 鍵は、RADIUS サーバ上で使用する暗号化鍵と照合する必要があるテキスト ストリングです。先行スペースは無視されますが、鍵の途中および末尾のスペースは使用されます。鍵にスペースを使用する場合は、鍵の一部として引用符を使用する場合を除いて、鍵を引用符で囲まないでください。
ステップ 3	<code>radius-server retransmit retries</code>	スイッチが、サーバに各 RADIUS 要求を送信する回数を指定します。デフォルトは 3 で、指定できる範囲は 1 ~ 1000 です。
ステップ 4	<code>radius-server timeout seconds</code>	スイッチが、RADIUS 要求に対する応答を待って要求を再送信するまでの秒数を指定します。デフォルトは 5 秒で、指定できる範囲は 1 ~ 1000 です。
ステップ 5	<code>radius-server deadtime minutes</code>	認証要求に応答しない RADIUS サーバをスキップする時間を指定します。これにより、要求がタイムアウトするまで待たずに、次の設定サーバを試行できます。デフォルトは 0 で、指定できる範囲は 1 ~ 1440 分です。
ステップ 6	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 7	<code>show running-config</code>	設定を確認します。
ステップ 8	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

再送信、タイムアウト、デッドタイムの設定をデフォルトに戻すには、これらのコマンドの **no** 形式を使用します。

## ベンダー固有の RADIUS アトリビュート用にスイッチを設定する方法

Internet Engineering Task Force (IETF) ドラフト規格では、ベンダー固有のアトリビュート (アトリビュート 26) を使用して、スイッチと RADIUS サーバとの間のベンダー固有情報の通信方式を定めています。Vendor-Specific Attribute (VSA) を使用すると、ベンダーは、汎用に適さない独自の拡張アトリビュートをサポートできます。シスコの実装 RADIUS では、仕様で推奨されたフォーマットを使用して 1 つのベンダー固有オプションをサポートします。シスコのベンダー ID は 9 で、サポート対照のオプションにはベンダータイプ 1 が設定されており、*cisco-avpair* と名前が付けられています。この値は次のフォーマットのストリングです。

```
protocol : attribute sep value *
```

*protocol* は、特定のタイプの許可に対応するシスコ プロトコル アトリビュートの値です。*attribute* と *value* は、シスコ TACACS+ 仕様で定義されている適正な AV のペアです。*sep* は、必須アトリビュートの場合は「=」、オプションのアトリビュートの場合は「\*」です。TACACS+ 許可で利用できるすべての機能は、RADIUS にも使用できます。

たとえば、次の AV ペアは、IP 許可時（PPP の IPCP アドレス割り当て時）に、シスコの複数の名前付き IP アドレスプール機能をアクティブにします。

```
cisco-avpair= "ip:addr-pool=first"
```

次に、スイッチからログインしているユーザに、イネーブル EXEC コマンドへの直接アクセスを可能にする例を示します。

```
cisco-avpair= "shell:priv-lvl=15"
```

次に、RADIUS サーバデータベース内の許可 VLAN を指定する例を示します。

```
cisco-avpair= "tunnel-type(#64)=VLAN(13)"
cisco-avpair= "tunnel-medium-type(#65)=802 media(6)"
cisco-avpair= "tunnel-private-group-ID(#81)=vlanid"
```

次に、この接続中に ASCII フォーマットの入力 ACL（アクセス制御リスト）をポートに適用する例を示します。

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any deernet-iv"
```

次に、この接続中に ASCII フォーマットの入力 ACL をポートに適用する例を示します。

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

その他のベンダーにも、独自に一意のベンダー ID、オプション、および対応する VSA が割り当てられます。ベンダー ID と VSA の詳細については、RFC 2138『Remote Authentication Dial-In User Service (RADIUS)』を参照してください。

VSA を認識して使用するようには、スイッチを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>radius-server vsa send [accounting   authentication]</code>	<p>スイッチが、RADIUS IETF アトリビュート 26 に定義されている VSA を認識して使用できるようにします。</p> <ul style="list-style-type: none"> <li>（任意）<b>accounting</b> キーワードを使用して、認識されるベンダー固有のアトリビュートの集合をアカウントング アトリビュートに限定します。</li> <li>（任意）<b>authentication</b> キーワードを使用して、認識されるベンダー固有のアトリビュートの集合を認証アトリビュートに限定します。</li> </ul> <p>キーワードなしでこのコマンドを入力すると、アカウントングおよび認証の両方のベンダー固有のアトリビュートが使用されます。</p>
ステップ 3	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 4	<code>show running-config</code>	設定を確認します。
ステップ 5	<code>copy running-config startup-config</code>	（任意）コンフィギュレーション ファイルに設定を保存します。


RADIUS アトリビュートの完全リスト、またはベンダー固有のアトリビュート 26 の詳細については、『Cisco IOS Security Configuration Guide』Release 12.2 の付録「RADIUS Attributes」を参照してください。

## ベンダー固有の RADIUS サーバ通信用にスイッチを設定する方法

RADIUS に関する IETF ドラフト規格では、スイッチと RADIUS サーバとの間のベンダー固有情報の通信方式を規定していますが、一部のベンダーは、固有の方法で RADIUS アトリビュートの集合を機能拡張しています。Cisco IOS ソフトウェアは、ベンダー固有仕様の RADIUS アトリビュートのサブセットをサポートします。

前述したように、RADIUS (ベンダー固有または IETF のドラフト準拠) を設定するには、RADIUS サーバデーモンが稼働しているホスト、およびスイッチと共有するシークレットテキストストリングを指定する必要があります。RADIUS ホストおよびシークレットテキストストリングを指定するには、**radius-server** グローバル コンフィギュレーション コマンドを使用します。

ベンダー固有の RADIUS サーバ ホスト、および共有シークレットテキストストリングを指定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<b>radius-server host {hostname   ip-address} non-standard</b>	リモート RADIUS サーバホストの IP アドレスまたはホスト名を指定し、ベンダー固有の実装 RADIUS を使用していることを明確にします。
ステップ 3	<b>radius-server key string</b>	<p>スイッチとベンダー固有の RADIUS サーバとの間で使用する、共有シークレットテキストストリングを指定します。スイッチおよび RADIUS サーバは、このテキストストリングを使用してパスワードを暗号化し、応答を交換します。</p> <p> (注) 鍵は、RADIUS サーバ上で使用する暗号化鍵と照合する必要があるテキストストリングです。先行スペースは無視されますが、鍵の途中および末尾のスペースは使用されます。鍵にスペースを使用する場合は、鍵の一部として引用符を使用する場合を除いて、鍵を引用符で囲まないでください。</p>
ステップ 4	<b>end</b>	イネーブル EXEC モードに戻ります。
ステップ 5	<b>show running-config</b>	設定を確認します。
ステップ 6	<b>copy running-config startup-config</b>	(任意) コンフィギュレーション ファイルに設定を保存します。

ベンダー固有の RADIUS ホストを削除するには、**no radius-server host {hostname | ip-address} non-standard** グローバル コンフィギュレーション コマンドを使用します。鍵をディセーブルにするには、**no radius-server key** グローバル コンフィギュレーション コマンドを使用します。

次に、ベンダー固有の RADIUS ホストを指定して、スイッチとサーバの間で *rad124* という秘密鍵を使用する例を示します。

```
Switch(config)# radius-server host 172.20.30.15 nonstandard
Switch(config)# radius-server key rad124
```

## RADIUS 設定の表示

RADIUS 設定情報を表示するには、**show running-config** イネーブル EXEC コマンドを使用します。



## Kerberos によるスイッチ アクセスの制御

ここでは、Kerberos セキュリティ システムをイネーブルにして設定する方法について説明します。Kerberos セキュリティ システムは、信頼できるサードパーティを使用してネットワーク リソースに対する要求を認証します。この機能を使用するには、スイッチ ソフトウェアの暗号化（暗号化をサポートする）バージョンがスイッチにインストールされている必要があります。この機能を使用し、Cisco.com から暗号化ソフトウェア ファイルをダウンロードするための許可を取得する必要があります。詳細については、現リリースのリリースノートを参照してください。

ここでは、次の内容について説明します。

- Kerberos の概要 (p.8-33)
- Kerberos の動作 (p.8-35)
- Kerberos の設定 (p.8-37)

Kerberos の設定例については、『Cisco IOS Security Configuration Guide』Release 12.2 の「Security Server Protocols」の章にある「Kerberos Configuration Examples」を参照してください。URLは次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fsecsp/](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/)



(注) ここで説明するコマンドの構文および使用方法の詳細については、『Cisco IOS Security Command Reference』Release 12.2 の「Security Server Protocols」の章にある「Kerberos Commands」を参照してください。URL は次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fsecsp/index.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/index.htm)



(注) Kerberos 構成例および『Cisco IOS Security Command Reference』Release 12.2 では、信頼できるサードパーティとして Catalyst 3750 Metro スイッチを使用しています。このスイッチは Kerberos に対応し、ネットワーク セキュリティ サーバとして設定可能で、Kerberos プロトコルを使用したユーザ認証ができます。

## Kerberos の概要

Kerberos は Massachusetts Institute of Technology (MIT) が開発した秘密鍵によるネットワーク認証プロトコルです。Data Encryption Standard (DES) という暗号化アルゴリズムを暗号化と認証に使用し、ネットワーク リソースに対する要求を認証します。Kerberos は、信頼できるサードパーティという概念を使ってユーザとサービスに対してセキュリティの検証を実行します。この信頼できるサードパーティを *Key Distribution Center* (KDC; 鍵発行局) と呼びます。

Kerberos は、ユーザが誰であるか、そのユーザが使用しているネットワーク サービスは何であるかを検証します。これを実行するために、KDC (つまり信頼できる Kerberos サーバ) がユーザにチケットを発行します。これらのチケットには有効期限があり、ユーザ証明書のキャッシュに保存されます。Kerberos サーバは、ユーザ名やパスワードの代わりにチケットを使ってユーザとネットワーク サービスを認証します。



(注) Kerberos サーバには、ネットワーク セキュリティ サーバとして設定されていて、Kerberos プロトコルを用いてユーザを認証できる Catalyst 3750 Metro スイッチを使用できます。

Kerberos の証明書発行スキームでは、*single logon* という手順を使用します。この手順では、ユーザを 1 回認証すると、ユーザ証明書が有効な間は（他のパスワードの暗号化を行わずに）セキュア認証が可能になります。

このソフトウェア リリースは Kerberos 5 に対応しています。Kerberos 5 では、すでに Kerberos 5 を使用している組織が、(UNIX サーバや PC などの) 他のネットワーク ホストが使用している KDC 上の Kerberos 認証データベースを使用できます。

このソフトウェア リリースでは、Kerberos は次のネットワーク サービスをサポートしています。

- Telnet
- rlogin
- Remote Shell Protocol (rsh)

表 8-2 に、一般的な Kerberos 関連用語とその定義を示します。

表 8-2 Kerberos の用語




用語	定義
認証	ユーザやサービスが他のサービスに対して自分自身の身元を証明する手順。たとえば、クライアントはスイッチに対して認証を得て、スイッチは他のスイッチに対して認証を得ます。
許可	ネットワークやスイッチにおいてユーザがどのような権限を有していて、またどのような動作を実行できるかを、スイッチが識別する手段。
証明書	認証チケット (Ticket Granting Ticket [TGT]、サービス証明書など) を表す総称。Kerberos 証明書で、ユーザまたはサービスの ID を検証します。ネットワークサービスがチケットを発行した Kerberos サーバを信頼することにした場合、ユーザ名やパスワードを再入力する代わりにこれを使用できます。証明書の有効期限は、8 時間がデフォルトの設定です。
インスタンス	Kerberos プリンシパルの認証レベル ラベル。ほとんどの Kerberos プリンシパルは、 <i>user@REALM</i> という形式です (たとえば、 <i>smith@EXAMPLE.COM</i> )。Kerberos インスタンスのある Kerberos プリンシパルは、 <i>user/instance@REALM</i> という形式です (たとえば、 <i>smith/admin@EXAMPLE.COM</i> )。Kerberos インスタンスは、認証が成功した場合のユーザの許可レベルを指定するのに使用できます。各ネットワーク サービスのサーバは、Kerberos インスタンスの許可マッピングを適用し実行できますが、必須ではありません。   (注) Kerberos プリンシパルとインスタンスの名前はすべて小文字でなければなりません。Kerberos レルム名はすべて大文字でなければなりません。
KDC	ネットワーク ホストで稼働する Kerberos サーバおよびデータベース プログラムで構成される KDC。
Kerberos 対応	Kerberos 証明書のインフラストラクチャをサポートするために変更されたアプリケーションやサービスのことを指す用語。
Kerberos レルム	Kerberos サーバに登録されたユーザ、ホスト、およびネットワーク サービスで構成されるドメイン。Kerberos サーバを信頼して、ユーザまたはネットワークサービスに対する別のユーザまたはネットワークサービスの ID を検証します。   (注) Kerberos レルム名はすべて大文字でなければなりません。

表 8-2 Kerberos の用語（続き）

用語	定義
Kerberos サーバ	ネットワーク ホストで稼働しているデーモン。ユーザおよびネットワーク サービスはそれぞれ Kerberos サーバに ID を登録します。ネットワーク サービスは Kerberos サーバにクエリを送信して、他のネットワーク サービスの認証を得ます。
KEYTAB	ネットワーク サービスが KDC と共有するパスワード。Kerberos 5 以降のバージョンでは、ネットワーク サービスは鍵テーブル (KEYTAB) を使用して暗号化されたサービス証明書を暗号解除して認証します。Kerberos 5 より前のバージョンでは、KEYTAB のことをサーバテーブル (SRVTAB) といいます。
プリンシパル	Kerberos ID と呼ばれ、Kerberos サーバに基づき、ユーザが誰であるか、サービスが何であるかを表します。  (注) Kerberos プリンシパル名はすべて小文字でなければなりません。
サービス証明書	ネットワーク サービスの証明書。KDC から証明書が発行されると、ネットワーク サービスと KDC が共有するパスワードで暗号化されます。ユーザ TGT とパスワードを共有します。
SRVTAB	ネットワーク サービスが KDC と共有するパスワード。Kerberos 5 以降のバージョンでは、SRVTAB は KEYTAB と呼びます。
TGT	身分証明書のことで、KDC が認証済みユーザに発行する証明書。TGT を受け取ったユーザは、KDC に示された Kerberos レalm内のネットワーク サービスに対して認証を得ることができます。

## Kerberos の動作

Kerberos サーバには、ネットワーク セキュリティ サーバとして設定されていて、Kerberos プロトコルを用いてリモートユーザを認証できる Catalyst 3750 Metro スイッチを使用できます。Kerberos をカスタマイズする方法はいくつかありますが、ネットワーク サービスにアクセスしようとするリモートユーザは、3 つのセキュリティ レイヤを通過しないとネットワーク サービスにアクセスできません。

Kerberos サーバとして Catalyst 3750 Metro スイッチを使用し、リモートユーザがネットワーク サービスに対して認証を得る手順は、次のとおりです。

1. 境界スイッチに対する認証の取得 (p.8-36)
2. KDC からの TGT の取得 (p.8-36)
3. ネットワークに対する認証の取得 (p.8-36)

## 境界スイッチに対する認証の取得

ここでは、リモート ユーザが通過しなければならない最初のセキュリティ レイヤについて説明します。ユーザは、まず境界スイッチに対して認証を得なければなりません。その後、このプロセスは次のように進みます。

1. ユーザが、境界スイッチへの Kerberos 非対応 Telnet 接続を確立します。
2. ユーザ名とパスワードの入力を求めるプロンプトをスイッチが表示します。
3. スイッチが、このユーザの TGT を KDC に要求します。
4. KDC がユーザ ID を含む暗号化された TGT をスイッチに送信します。
5. スイッチは、ユーザが入力したパスワードを使って TGT の暗号解除を試行します。
  - 暗号解除に成功したら、ユーザはスイッチに対して認証を得ます。
  - 暗号解除に成功しなかった場合、ユーザは、ユーザ名とパスワードを再入力するか (Caps Lock または Num Lock のオン/オフに注意)、別のユーザ名とパスワードを入力してステップ 2 をやり直します。

境界スイッチに対して Kerberos 非対応 Telnet セッションを確立し認証を得たリモート ユーザは、ファイアウォールの内側にいますが、ネットワーク サービスへのアクセス権を取得するために、さらに KDC に対して直接認証を得る必要があります。ユーザが KDC から認証を得なければならないのは、KDC が発行する TGT はスイッチに保存されていて、ユーザがスイッチにログオンしないと追加の認証に使えないからです。

## KDC からの TGT の取得

ここでは、リモート ユーザが通過しなければならない 2 番めのセキュリティ レイヤについて説明します。ユーザは、ネットワークにアクセスするために、ここで KDC の認証を得て KDC から TGT を取得しなければなりません。

KDC に対して認証を得る方法については、『Cisco IOS Security Configuration Guide』Release 12.2 の「Security Server Protocols」の章にある「Obtaining a TGT from a KDC」を参照してください。URL は次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fseccsp/scfkerb.htm#1000999](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fseccsp/scfkerb.htm#1000999)

## ネットワークに対する認証の取得

ここでは、リモート ユーザが通過しなければならない 3 番めのセキュリティ レイヤについて説明します。TGT を持つユーザは、ここで Kerberos レalm内のネットワーク サービスに対して認証を得なければなりません。

ネットワーク サービスに対して認証を得る方法については、『Cisco IOS Security Configuration Guide』Release 12.2 の「Security Server Protocols」の章にある「Authenticating to Network Services」を参照してください。URL は次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fseccsp/scfkerb.htm#1001010](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fseccsp/scfkerb.htm#1001010)

## Kerberos の設定

リモート ユーザがネットワーク サービスに対して認証を得られるように、Kerberos レalm内のホストと KDC を設定して、ユーザおよびネットワーク サービスと通信を行って互いに認証するようになければなりません。これを実行するために、互いを識別しなければなりません。KDC 上の Kerberos データベースにホストのエントリを追加し、Kerberos レalm内のすべてのホストに KDC が生成した KEYTAB ファイルを追加します。また、KDC データベースにユーザ用のエントリも作成します。

ホストおよびユーザのエントリを追加または作成する場合は、次の注意事項に従ってください。

- Kerberos プリンシパル名はすべて小文字でなければなりません。
- Kerberos インスタンス名はすべて小文字でなければなりません。
- Kerberos レalm名はすべて大文字でなければなりません。



(注) Kerberos サーバには、ネットワーク セキュリティ サーバとして設定されていて、Kerberos プロトコルを用いてユーザを認証できる Catalyst 3750 Metro スイッチを使用できます。

Kerberos 認証済みサーバクライアント システムを設定する手順は、次のとおりです。

- Kerberos コマンドを使用して KDC を設定します。
- Kerberos プロトコルを使用するようにスイッチを設定します。

設定については、『Cisco IOS Security Configuration Guide』 Release 12.2 の「Security Server Protocols」の章にある「Kerberos Configuration Task List」を参照してください。URL は次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fsecsp/scfkerb.htm#1001027](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fsecsp/scfkerb.htm#1001027)

## スイッチのローカル認証および許可の設定

ローカル モードで AAA を実装するようにスイッチを設定すると、サーバがなくても AAA が動作するように設定できます。この場合、スイッチが認証および許可の処理を行います。この設定ではアカウント機能は利用できません。

スイッチをローカル AAA 用に設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>aaa new-model</code>	AAA をイネーブルにします。
ステップ 3	<code>aaa authentication login default local</code>	ローカルのユーザ名データベースを使用するようにログイン認証を設定します。 <b>default</b> キーワードにより、ローカル ユーザ データベース認証がすべてのポートに適用されます。
ステップ 4	<code>aaa authorization exec local</code>	ユーザ AAA 許可を設定し、ローカル データベースをチェックし、ユーザが EXEC シェルを実行できるようにします。
ステップ 5	<code>aaa authorization network local</code>	ネットワーク関連のすべてのサービス要求に関する AAA 許可を設定します。
ステップ 6	<code>username name [privilege level] {password encryption-type password}</code>	ローカル データベースを開始し、ユーザ名ベースの認証システムを設定します。  ユーザごとにこのコマンド入力を繰り返します。 <ul style="list-style-type: none"> <li><i>name</i> には、ユーザ ID を 1 ワードで指定します。スペースや引用符は使用できません。</li> <li>(任意) <i>level</i> には、アクセス後ユーザに設定する権限レベルを指定します。指定できる範囲は 0 ~ 15 です。レベル 15 ではイネーブル EXEC モードでのアクセスが可能です。レベル 0 では、ユーザ EXEC モードでのアクセスとなります。</li> <li><i>encryption-type</i> には、暗号化されていないパスワードがあとに続く場合は 0 を、暗号化されたパスワードがあとに続く場合は 7 を指定します。</li> <li><i>password</i> には、ユーザがスイッチにアクセスする場合に入力する必要のあるパスワードを指定します。パスワードは 1 ~ 25 文字で、埋め込みスペースを使用でき、<b>username</b> コマンドの最後のオプションとして指定します。</li> </ul>
ステップ 7	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 8	<code>show running-config</code>	設定を確認します。
ステップ 9	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

AAA をディセーブルにするには、**no aaa new-model** グローバル コンフィギュレーション コマンドを使用します。許可をディセーブルにするには、**no aaa authorization {network | exec} method1** グローバル コンフィギュレーション コマンドを使用します。

## SSH のためのスイッチの設定

ここでは、SSH 機能を設定する方法について説明します。この機能を使用するには、スイッチに暗号化ソフトウェア イメージをインストールする必要があります。この機能を使用し、Cisco.com から暗号化ソフトウェア ファイルをダウンロードするための許可を取得する必要があります。詳細については、現リリースのリリースノートを参照してください。

ここでは、次の情報について説明します。

- SSH の概要 (p.8-39)
- SSH の設定 (p.8-40)
- SSH コンフィギュレーションおよびステータスの表示 (p.8-43)

SSH の設定例については、『Cisco IOS Security Configuration Guide, Cisco IOS Release 12.2』の「Configuring Secure Shell」の章にある「SSH Configuration Examples」を参照してください。URL は次のとおりです。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_c/fothersf/scfssh.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_c/fothersf/scfssh.htm)



(注)

この章で使用されるコマンドの構文および使用方法の詳細については、このリリースのコマンドリファレンス、および Cisco IOS Release 12.2 のコマンドリファレンスを参照してください。URL は次のとおりです。

<http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/index.htm>

## SSH の概要

SSH は、デバイスへの安全なリモート接続を提供するプロトコルです。SSH は、デバイスの認証時に強力な暗号化を行うことで、リモート接続について Telnet 以上のセキュリティを実現します。このソフトウェア リリースは、SSH バージョン 1 (SSHv1) および SSH バージョン 2 (SSHv2) をサポートします。

ここでは、次の内容について説明します。

- SSH サーバ、統合クライアント、およびサポート対象バージョン (p.8-39)
- 制限事項 (p.8-40)

## SSH サーバ、統合クライアント、およびサポート対象バージョン

SSH 機能には、スイッチで稼働するアプリケーションである、SSH サーバおよび SSH 統合クライアントがあります。SSH サーバが稼働するスイッチとの接続には、SSH クライアントを使用できます。SSH サーバは、このリリースでサポートされる SSH クライアントおよび他社製の SSH クライアントと連動します。また SSH クライアントも、このリリースでサポートされる SSH サーバおよび他社製の SSH サーバと連動します。

このスイッチは、SSHv1 または SSHv2 サーバをサポートします。

このスイッチは、SSHv1 クライアントをサポートします。

SSH は、Data Encryption Standard (DES; データ暗号化規格) 暗号化アルゴリズム、トリプル DES (3DES) 暗号化アルゴリズム、パスワードベースのユーザ認証をサポートします。

SSH は、次のユーザ認証方式もサポートします。

- TACACS+ (詳細については、「[TACACS+ によるスイッチアクセスの制御](#)」 [p.8-11] を参照)
- RADIUS (詳細については、「[RADIUS によるスイッチアクセスの制御](#)」 [p.8-19] を参照)
- ローカル認証および許可 (詳細については、「[スイッチのローカル認証および許可の設定](#)」 [p.8-38] を参照)



(注) このソフトウェアリリースでは、IP Security (IPSec) をサポートしていません。

## 制限事項

SSH には、次の制限事項が適用されます。

- スイッチは、Rivest, Shamir, and Adelman (RSA) 認証をサポートします。
- SSH は、シェル実行アプリケーションだけをサポートします。
- SSH サーバおよび SSH クライアントは、DES (56 ビット) および 3DES (168 ビット) のデータ暗号化ソフトウェアでのみサポートされます。
- スイッチは、Advanced Encryption Standard (AES) 対称暗号化アルゴリズムをサポートしません。

## SSH の設定

ここでは、次の設定情報について説明します。

- [設定時の注意事項](#) (p.8-40)
- [スイッチでの SSH 実行の設定](#) (p.8-41) (必須)
- [SSH サーバの設定](#) (p.8-42) (スイッチを SSH サーバとして設定している場合にのみ必要)

## 設定時の注意事項

スイッチを SSH サーバまたは SSH クライアントとして設定するときは、次の注意事項に従ってください。

- SSHv1 サーバにより生成された RSA 鍵ペアは、SSHv2 サーバで使用できます。また、その逆も可能です。
- **crypto key generate rsa** グローバル コンフィギュレーション コマンドを入力したあと、CLI エラーメッセージが表示される場合、RSA 鍵ペアは生成されていません。ホスト名およびドメインを再設定してから、**crypto key generate rsa** コマンドを入力してください。詳細については、「[スイッチでの SSH 実行の設定](#)」 (p.8-41) を参照してください。
- RSA 鍵ペアを生成するときに、「No host name specified」というメッセージが表示される場合があります。表示される場合は、**hostname** グローバル コンフィギュレーション コマンドを使用して、ホスト名を設定する必要があります。
- RSA 鍵ペアを生成するときに、「No domain specified」というメッセージが表示される場合があります。表示される場合は、**ip domain-name** グローバル コンフィギュレーション コマンドを使用して、IP ドメイン名を設定する必要があります。
- ローカルの AAA 方式を設定する場合は、コンソール上で AAA がディセーブルであることを確認してください。



## スイッチでの SSH 実行の設定

スイッチに SSH の実行を設定するには、次の手順を実行します。

1. Cisco.com から暗号化ソフトウェア イメージをダウンロードします。この手順は必須です。詳細については、現リリースのリリースノートを参照してください。
2. スイッチに、ホスト名および IP ドメイン名を設定します。この手順は、スイッチを SSH サーバとして設定している場合にのみ実行します。
3. スイッチに RSA 鍵ペアを生成します。これにより自動的に SSH がイネーブルになります。この手順は、スイッチを SSH サーバとして設定している場合にのみ実行します。
4. ローカルまたはリモート アクセスにユーザ認証を設定します。この手順は必須です。詳細については、「[スイッチのローカル認証および許可の設定](#)」(p.8-38) を参照してください。

ホスト名および IP ドメイン名を設定して、RSA 鍵ペアを生成するには、イネーブル EXEC モードで次の手順を実行します。この手順は、スイッチを SSH サーバとして設定している場合に必要です。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>hostname hostname</code>	スイッチにホスト名を設定します。
ステップ 3	<code>ip domain-name domain_name</code>	スイッチにホスト ドメインを設定します。
ステップ 4	<code>crypto key generate rsa</code>	スイッチのローカルおよびリモート認証に関して、SSH サーバをイネーブルにして、RSA 鍵ペアを生成します。  モジュール サイズを 1024 ビット以上にすることを推奨します。  RSA 鍵ペアを生成すると、モジュールの長さを入力するよう求められます。モジュールの長さが長い方がセキュリティは高くなりますが、生成および使用に時間がかかります。
ステップ 5	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 6	<code>show ip ssh</code>  または <code>show ssh</code>	SSH サーバのバージョンおよび設定情報を表示します。  スイッチ上の SSH サーバのステータスを表示します。
ステップ 7	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

RSA 鍵ペアを削除するには、`crypto key zeroize rsa` グローバル コンフィギュレーション コマンドを使用します。RSA 鍵ペアを削除すると、SSH サーバは自動的にディセーブルになります。

## SSH サーバの設定

SSH サーバを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>ip ssh version [1   2]</code>	<p>(任意) スイッチに SSH バージョン 1 または 2 を実行するよう設定します。</p> <ul style="list-style-type: none"> <li>• 1 — スイッチに SSH バージョン 1 を実行するよう設定します。</li> <li>• 2 — スイッチに SSH バージョン 2 を実行するよう設定します。</li> </ul> <p>このコマンドを入力しない場合、またはキーワードを指定しない場合は、SSH サーバにより SSH クライアントがサポートする最新の SSH バージョンが選択されます。たとえば、SSH クライアントが SSHv1 および SSHv2 をサポートしている場合、SSH サーバは SSHv2 を選択します。</p>
ステップ 3	<code>ip ssh {timeout seconds   authentication-retries number}</code>	<p>SSH 制御パラメータを設定します。</p> <ul style="list-style-type: none"> <li>• タイムアウト値を秒単位で指定します。デフォルトでは 120 秒です。指定できる範囲は 0 ~ 120 秒です。このパラメータは、SSH ネゴシエーションフェーズに適用されます。接続が確立されると、スイッチはデフォルトの CLI ベースセッションのタイムアウト値を使用します。</li> </ul> <p>デフォルトでは、複数の CLI ベースセッションに関する暗号化 SSH 接続を最大 5 つまで同時にネットワーク上で使用できます (セッション 0 からセッション 4)。シェル実行が開始されると、CLI ベースセッションのタイムアウト値がデフォルトの 10 分に戻ります。</p> <ul style="list-style-type: none"> <li>• クライアントがサーバに対して再認証できる回数を指定します。デフォルトは 3 で、指定できる範囲は 0 ~ 5 です。</li> </ul> <p>両方のパラメータを設定する場合は、このステップを繰り返します。</p>
ステップ 4	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 5	<code>show ip ssh</code>  または <code>show ssh</code>	<p>SSH サーバのバージョンおよび設定情報を表示します。</p> <p>スイッチ上の SSH サーバ接続のステータスを表示します。</p>
ステップ 6	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

デフォルトの SSH 制御パラメータに戻るには、`no ip ssh {timeout | authentication-retries}` グローバル コンフィギュレーション コマンドを使用します。

## SSH コンフィギュレーションおよびステータスの表示

SSH サーバの設定およびステータスを表示するには、表 8-3 に示すイネーブル EXEC コマンドを 1 つまたは複数使用します。

**表 8-3 SSH サーバの設定およびステータスを表示するコマンド**

コマンド	説明
<code>show ip ssh</code>	SSH サーバのバージョンおよび設定情報を表示します。
<code>show ssh</code>	SSH サーバのステータスを表示します。

これらのコマンドの詳細については、『*Cisco IOS Security Command Reference, Cisco IOS Release 12.2*』の「Other Security Features」の章の「Secure Shell Commands」を参照してください。次の URL からアクセスしてください。

[http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur\\_r/fothercr/srfssh.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fsecur_r/fothercr/srfssh.htm)

## スイッチの SSL HTTP の設定

ここでは、HTTP 1.1 サーバおよびクライアント用の Secure Socket Layer (SSL) バージョン 3.0 サポートの設定方法について説明します。SSL は、サーバ認証、暗号化、メッセージ完全性、および HTTP クライアント認証を提供し、安全な HTTP 通信を可能にします。この機能を使用するには、スイッチに暗号化ソフトウェア イメージをインストールする必要があります。この機能を使用し、Cisco.com から暗号化ソフトウェア ファイルをダウンロードするための許可を取得する必要があります。暗号化イメージの詳細については、現リリースのリリースノートを参照してください。

ここでは、次の情報について説明します。

- セキュア HTTP サーバおよびクライアントの概要 (p.8-44)
- セキュア HTTP サーバおよびクライアントの設定 (p.8-46)
- セキュア HTTP サーバおよびクライアントのステータス表示 (p.8-50)

ここで説明する設定例と、コマンドの構文および使用方法の詳細については、Cisco IOS Release 12.2(15)T の「HTTPS - HTTP Server and Client with SSL 3.0」の機能説明を参照してください。URL は次のとおりです。

<http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t15/ftsslsh.htm>

## セキュア HTTP サーバおよびクライアントの概要

セキュア HTTP 接続では、HTTP サーバ間でやり取りされるデータはインターネット上で送信される前に暗号化されます。SSL 暗号化を含む HTTP では安全な接続が提供され、Web ブラウザからスイッチを設定する機能などが可能になります。シスコのセキュア HTTP サーバとセキュア HTTP クライアントの実装では、アプリケーション レイヤ暗号化を使用した SSL バージョン 3.0 の実装が使用されます。HTTP over SSL は HTTPS と呼ばれます。安全な接続の URL は `http://` ではなく `https://` から始まります。

HTTP セキュア サーバ (スイッチ) の主な役割は、指定ポート (デフォルトの HTTPS ポートは 443) で HTTPS 要求を待ち受けて、要求を HTTP 1.1 Web サーバに渡すことです。HTTP 1.1 サーバが要求を処理し、応答 (ページ) を HTTP セキュア サーバに返します。HTTP セキュア サーバは元の要求に応答します。

HTTP セキュア クライアント (Web ブラウザ) の主な役割は、HTTPS ユーザ エージェント サービスの Cisco IOS アプリケーション要求に応答し、アプリケーションの HTTPS User Agent サービスを実行し、アプリケーションに応答を返すことです。

## 認証局トラストポイント

Certificate Authority (CA; 認証局) は、証明書要求を管理して、関係するネットワーク デバイスに証明書を発行します。このサービスにより、中央集中型セキュリティ鍵および証明書管理が関係デバイスに提供されます。特定の CA サーバはトラストポイントと呼ばれます。

接続を試行すると、HTTPS サーバが指定された CA トラストポイントから取得した認証済みの X.509.3 証明書を発行して、クライアントに安全な接続を提供します。クライアント (通常は Web ブラウザ) には公開鍵があり、証明書を認証できるようになっています。

セキュア HTTP 接続の場合、CA トラストポイントを設定することを強く推奨します。CA トラストポイントが HTTPS サーバを稼働しているデバイスに設定されていない場合、サーバは自身を認証し必要な RSA 鍵ペアを生成します。自己認証 (自己署名) 証明書ではセキュリティが十分ではないので、接続するクライアントは、証明書が自己認証されていることをユーザに通知し、ユーザは接続を許可または拒否します。このオプションは、内部ネットワーク トポロジー (テストなど) で便利です。

CA トラストポイントを設定しない場合、セキュア HTTP 接続をイネーブルにすると、セキュア HTTP サーバ（またはクライアント）に一時または持続自己署名証明書が自動的に生成されます。

- スイッチにホスト名やドメイン名が設定されていない場合、一時自己署名証明書が生成されず。スイッチを再起動すると、一時自己署名証明書は失われ、新しい一時自己署名証明書が割り当てられます。
- スイッチにホスト名やドメイン名が設定されている場合、持続自己署名証明書が生成されず。この証明書は、スイッチを再起動したりセキュア HTTP サーバをディセーブルにしてもアクティブなので、次にセキュア HTTP 接続を再びイネーブルにしても存在します。

自己署名証明書が生成されたら、この情報には **show running-config** イネーブル EXEC コマンドの出力が含まれます。次に、自己署名証明書を表示するコマンドからの出力例の一部を示します。

```
Switch# show running-config
Building configuration...
```

(テキスト出力は省略)

```
crypto pki trustpoint TP-self-signed-3080755072
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-3080755072
  revocation-check none
  rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
  certificate self-signed 01
    3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
    59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
    69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
    02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
    30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D
```

(テキスト出力は省略)

セキュア HTTP サーバをディセーブルにし、**no crypto pki trustpoint TP-self-signed-30890755072** グローバル コンフィギュレーション コマンドを入力すると、この自己署名証明書を削除できます。その後、セキュア HTTP サーバを再びイネーブルにする場合、新しい自己署名証明書が生成されず。



(注) *TP self-signed* 以降の値は、デバイスのシリアル番号によって異なります。

オプションのコマンド (**ip http secure-client-auth**) を使用すると、HTTPS サーバはクライアントから X.509v3 証明書を要求できます。クライアントを認証すると、サーバ認証のみよりもセキュリティが高くなります。

認証局の詳細については、『Cisco IOS Security Configuration Guide』 Release 12.2 の「Configuring Certification Authority Interoperability」の章の「Certificate Authorities」を参照してください。

## CipherSuite

CipherSuite では、SSL 接続で使用するための暗号化アルゴリズムおよびダイジェストアルゴリズムが指定されます。HTTPS サーバに接続する場合、クライアント Web ブラウザはサポートされた CipherSuite リストを提供し、クライアントおよびサーバは、両方サポートされているリスト上の CipherSuite から最適な暗号化アルゴリズムをネゴシエーションします。たとえば、Netscape Communicator 4.76 では、U.S. security with RSA Public Key Cryptography、MD2、MD5、RC2-CBC、RC4、DES-CBC、DES-EDE3-CBC をサポートします。

最適な暗号化のためには、Microsoft Internet Explorerバージョン 5.5以降または Netscape Communicator version 4.76 以降などの、128 ビット暗号化をサポートするクライアント ブラウザを使用する必要があります。SSL\_RSA\_WITH\_DES\_CBC\_SHA CipherSuite は、128 ビット暗号化を使用しないので他の CipherSuites よりもセキュリティの点で劣ります。

セキュリティがより高く複雑な CipherSuite では、処理時間がわずかに長くなります。次のリストでは、スイッチでサポートされた CipherSuite を定義し、ルータ処理負荷（速度）の点で速いものから遅いものに順位付けします。

1. SSL\_RSA\_WITH\_DES\_CBC\_SHA — メッセージの暗号化の場合は DES-CBC、メッセージダイジェストの場合は SHA による RSA 鍵交換（RSA 公開鍵暗号化）
2. SSL\_RSA\_WITH\_RC4\_128\_MD5 — RC4 128 ビット暗号化およびメッセージダイジェストの MD5 による RSA 鍵交換
3. SSL\_RSA\_WITH\_RC4\_128\_SHA — RC4 128 ビット暗号化およびメッセージダイジェストの SHA による RSA 鍵交換
4. SSL\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA — メッセージの暗号化の場合は 3DES および DES-EDE3-CBC、メッセージダイジェストの場合は SHA による RSA 鍵交換

SSL 接続での鍵生成および認証の両方には、RSA（および指定された暗号化とダイジェストアルゴリズムの組み合わせ）が使用されます。CA トラストポイントが設定されているかどうかに関係なく、RSA が使用されます。

## セキュア HTTP サーバおよびクライアントの設定

ここでは、HTTP サーバおよびクライアントで SSL を設定する手順について説明します。内容は次のとおりです。

- [SSL のデフォルト設定 \(p.8-46\)](#)
- [SSL 設定時の注意事項 \(p.8-47\)](#)
- [CA トラストポイントの設定 \(p.8-47\)](#)
- [セキュア HTTP サーバの設定 \(p.8-48\)](#)
- [セキュア HTTP クライアントの設定 \(p.8-50\)](#)

## SSL のデフォルト設定

標準 HTTP サーバはイネーブルです。

SSL はイネーブルです。

CA トラストポイントは設定されていません。

自己署名証明書は生成されません。

## SSL 設定時の注意事項

スイッチ クラスタで SSL を使用する場合、SSL セッションはクラスタ コマンドで終了します。クラスタ メンバースイッチは標準 HTTP で動作する必要があります。

CA トラストポイントを設定する前に、システム クロックが設定されていることを確認します。クロックが設定されていない場合は、日付が正しくないために証明書が拒否されます。

## CA トラストポイントの設定

セキュア HTTP 接続の場合、公式 CA トラストポイントを設定することを推奨します。CA トラストポイントは、自己署名証明書より安全です。

CA トラストポイントを設定するには、イネーブル EXEC モードで次の手順を実行します。


	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 2	<code>hostname hostname</code>	スイッチのホスト名を指定します (ホスト名を事前に設定していない場合のみ必須)。ホスト名は、セキュリティ鍵と証明書で必要となります。
ステップ 3	<code>ip domain-name domain-name</code>	スイッチの IP ドメイン名を指定します (IP ドメイン名を事前に設定していない場合のみ必須)。IP ドメイン名は、セキュリティ鍵と証明書で必要となります。
ステップ 4	<code>crypto key generate rsa</code>	(任意) RSA 鍵のペアを生成します。RSA 鍵のペアは、スイッチの証明書を取得する前に必要となります。RSA 鍵のペアは自動的に生成されます。必要な場合は、このコマンドを使用して鍵を再生成できます。
ステップ 5	<code>crypto ca trustpoint name</code>	CA トラストポイントのローカル設定名を指定し、CA トラストポイント設定モードを開始します。
ステップ 6	<code>enrollment url url</code>	スイッチが証明書要求を送信する相手先 URL を指定します。
ステップ 7	<code>enrollment http-proxy host-name port-number</code>	(任意) HTTP プロキシサーバによって CA から証明書を取得するようスイッチを設定します。
ステップ 8	<code>crl query url</code>	Certificate Revocation List (CRL; 証明書失効リスト) を要求して、ペアの証明書が失効していないことを確認するようにスイッチを設定します。
ステップ 9	<code>primary</code>	(任意) CA 要求のプライマリ (デフォルト) トラストポイントとしてトラストポイントを使用することを指定します。
ステップ 10	<code>exit</code>	CA トラストポイント設定モードを終了し、グローバル コンフィギュレーション モードに戻ります。
ステップ 11	<code>crypto ca authentication name</code>	CA の公開鍵を取得して CA を認証します。ステップ 5 で使用した名前を使用します。
ステップ 12	<code>crypto ca enroll name</code>	指定した CA トラストポイントから証明書を取得します。このコマンドでは、RSA 鍵のペアごとに署名された証明書を要求します。
ステップ 13	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 14	<code>show crypto ca trustpoints</code>	設定を確認します。
ステップ 15	<code>copy running-config startup-config</code>	(任意) コンフィギュレーション ファイルに設定を保存します。

CA に関連付けられたすべてのアイデンティティ情報および証明書を削除するには `no crypto ca trustpoint name` グローバル コンフィギュレーション コマンドを使用します。

## セキュア HTTP サーバの設定

証明書に CA を使用している場合、前の手順を実行してスイッチに CA トラストポイントを設定してから HTTP サーバをイネーブルにする必要があります。CA トラストポイントを設定しない場合、セキュア HTTP サーバを初めてイネーブルにしたとき、自己署名証明書が生成されます。サーバを設定したあとで、標準 HTTP サーバおよびセキュア HTTP サーバの両方に適用されるオプション (パス、適用するアクセスリスト、接続の最大数、またはタイムアウトポリシー) を設定できます。

セキュア HTTP サーバを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>show ip http server status</code>	(任意) HTTP サーバのステータスを表示し、セキュア HTTP サーバ機能がソフトウェアでサポートされているかどうかを判断します。次の行のうちいずれかが出力で表示される必要があります。  HTTP secure server capability: Present または HTTP secure server capability: Not present
ステップ 2	<code>configure terminal</code>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<code>ip http secure-server</code>	HTTPS サーバがディセーブルになっている場合はイネーブルにします。HTTPS サーバはデフォルトではイネーブルに設定されています。
ステップ 4	<code>ip http secure-port port-number</code>	(任意) HTTPS サーバに使用するポート番号を指定します。デフォルトのポート番号は 443 です。有効なオプションは、443 または 1025 ~ 65535 までの任意の数値です。
ステップ 5	<code>ip http secure-ciphersuite</code> {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}	(任意) HTTPS 接続で暗号化に使用する CipherSuite (暗号化アルゴリズム) を指定します。特定の CipherSuite を指定する理由がない場合、サーバおよびクライアントが、両方でサポートされている CipherSuite をネゴシエーションできるようにする必要があります。これはデフォルト設定です。
ステップ 6	<code>ip http secure-client-auth</code>	(任意) 接続プロセス中にクライアントから認証用の X.509v3 証明書を要求するように HTTP サーバを設定します。デフォルトでは、クライアントがサーバから証明書を要求しますが、サーバはクライアントを認証しようとしません。
ステップ 7	<code>ip http secure-trustpoint name</code>	X.509v3 セキュリティ証明書を取得し、クライアント証明書接続を認証するのに使用する CA トラストポイントを指定します。   (注) このコマンドを使用する場合、前の手順に従って CA トラストポイントをすでに設定していることを前提としています。
ステップ 8	<code>ip http path path-name</code>	(任意) HTML ファイルのベース HTTP パスを設定します。このパスでは、ローカル システム上の HTTP サーバ ファイルの場所を指定します (通常の場合はシステムのフラッシュ メモリ内にあります)。
ステップ 9	<code>ip http access-class access-list-number</code>	(任意) HTTP サーバへのアクセスを許可するアクセス リストを指定します。
ステップ 10	<code>ip http max-connections value</code>	(任意) HTTP サーバに許可される同時接続の最大数を設定します。指定できる範囲は 1 ~ 16 で、デフォルト値は 5 です。



	コマンド	説明
ステップ 11	<code>ip http timeout-policy idle seconds life seconds requests value</code>	(任意) HTTP サーバへの接続が、規定の状況下でオープンの状態であることができる時間を指定します。 <ul style="list-style-type: none"> <li>• <b>idle</b> — データを受信しないか応答データを送信できない最長時間。指定できる範囲は 1 ~ 600 秒です。デフォルト値は 180 秒 (3 分) です。</li> <li>• <b>life</b> — 接続が確立したときからの最長時間。指定できる範囲は 1 ~ 86400 秒 (24 時間) です。デフォルト値は 180 秒です。</li> <li>• <b>requests</b> — 持続した接続で処理される要求の最大数。最大値は 86400 です。デフォルトは 1 です。</li> </ul>
ステップ 12	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 13	<code>show ip http server secure status</code>	HTTP セキュア サーバのステータスを表示し、設定を確認します。
ステップ 14	<code>copy running-config startup-config</code>	(任意) コンフィギュレーションファイルに設定を保存します。

標準 HTTP サーバをディセーブルにするには、`no ip http server` グローバル コンフィギュレーション コマンドを使用します。セキュア HTTP サーバをディセーブルにするには、`no ip http secure-server` グローバル コンフィギュレーション コマンドを使用します。デフォルト設定に戻すには `no ip http secure-port` および `no ip http secure-ciphersuite` グローバル コンフィギュレーション コマンドを使用します。クライアント認証の要求を削除するには、`no ip http secure-client-auth` グローバル コンフィギュレーション コマンドを使用します。

Web ブラウザを使用してセキュア HTTP 接続を確認するには、`https://URL` と入力します。URL は、サーバスイッチの IP アドレスまたはホスト名です。デフォルト ポート以外のポートを設定した場合は、URL のあとにポート番号を指定する必要もあります。次に例を示します。

```
https://209.165.129:1026
```

または

```
https://host.domain.com:1026
```

## セキュア HTTP クライアントの設定

標準 HTTP クライアントおよびセキュア HTTP クライアントは、常にイネーブルです。セキュア HTTP クライアント証明書には CA が必要です。次の手順では、CA トラストポイントをスイッチで設定してあることを前提にしています。CA トラストポイントが設定されてなく、リモート HTTPS サーバがクライアント認証を要求する場合、セキュア HTTP クライアントへの接続は失敗となります。

セキュア HTTP クライアントを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	説明
ステップ 1	<code>configure terminal</code>	グローバル コンフィギュレーションモードを開始します。
ステップ 2	<code>ip http client secure-trustpoint name</code>	(任意) リモート HTTP サーバがクライアント認証を要求した場合に使用する CA トラストポイントを指定します。このコマンドを使用する場合、前の手順に従って CA トラストポイントをすでに設定していることを前提にしています。クライアント認証が必要ない場合、またはプライマリ トラストポイントが設定されている場合、このコマンドの実行は任意です。
ステップ 3	<code>ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}</code>	(任意) HTTPS 接続で暗号化に使用する CipherSuite (暗号化アルゴリズム) を指定します。特定の CipherSuite を指定する理由がない場合、サーバおよびクライアントが、両方でサポートされている CipherSuite をネゴシエーションできるようにする必要があります。これはデフォルト設定です。
ステップ 4	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 5	<code>show ip http client secure status</code>	HTTP セキュア サーバのステータスを表示し、設定を確認します。
ステップ 6	<code>copy running-config startup-config</code>	(任意) コンフィギュレーションファイルに設定を保存します。

クライアントのトラストポイント設定を削除するには、`no ip http client secure-trustpoint name` を使用します。クライアントに以前設定した CipherSuite 指定を削除するには、`no ip http client secure-ciphersuite` を使用します。

## セキュア HTTP サーバおよびクライアントのステータス表示

SSL セキュア サーバおよびクライアントのステータスを表示するには、表 8-4 のイネーブル EXEC コマンドを使用します。

表 8-4 SSL セキュア サーバおよびクライアントのステータスを表示するコマンド

コマンド	説明
<code>show ip http client secure status</code>	HTTP セキュア クライアント設定を表示します。
<code>show ip http server secure status</code>	HTTP セキュア サーバ設定を表示します。
<code>show running-config</code>	セキュア HTTP 接続用に生成された自己署名証明書を表示します。