



# スイッチベースの認証の設定

---

この章では、Catalyst 3550 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 \(p.7-2\)](#)
- [イネーブル EXEC コマンドへのアクセスの保護 \(p.7-3\)](#)
- [TACACS+ によるスイッチ アクセスの制御 \(p.7-12\)](#)
- [RADIUS によるスイッチ アクセスの制御 \(p.7-20\)](#)
- [Kerberos によるスイッチ アクセスの制御 \(p.7-34\)](#)
- [スイッチのローカル認証および許可の設定 \(p.7-39\)](#)
- [SSH のためのスイッチの設定 \(p.7-40\)](#)
- [SSL HTTP のためのスイッチの設定 \(p.7-44\)](#)
- [Secure Copy Protocol のためのスイッチの設定 \(p.7-51\)](#)

## 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の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.7-3)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インターフェイスに割り当てられ、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。イネーブル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イネーブルレベル(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.7-8)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ング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.7-12)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イネーブル 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。イネーブル レベルによって、ネットワーク デバイスにログオンしたあと、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 for Cisco IOS』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イネーブル レベル設定 (p.7-3)
- 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.7-4)
- 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.7-5)
- パスワード回復のディセーブル化 (p.7-6)
- 端末回線に対する Telnet パスワードの設定 (p.7-7)
- ユーザ名とパスワードのペアの設定 (p.7-8)
- 複数のイネーブル レベルの設定 (p.7-9)

### デフォルトのパスワードおよびイネーブル レベル設定

表 7-1 に、デフォルトのパスワードおよびイネーブル レベル設定を示します。

表 7-1 デフォルトのパスワードおよびイネーブル レベル

機能	デフォルト設定
イネーブル パスワードおよびイネーブル 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 パスワードおよびイネーブル 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	目的
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>	<p>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。</p> <p>デフォルトでは、パスワードは定義されていません。</p> <p><i>password</i> には、1 ~ 25 文字の英数字の文字列を指定します。文字列は数字で始めることはできません。大文字と小文字が区別され、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。</p> <p><code>abc</code> を入力します。</p> <p><code>Ctrl-v</code> を入力します。</p> <p><code>?123</code> を入力します。</p> <p>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。</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>	<p>(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。</p> <p>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。</p>

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

追加のセキュリティレイヤを、特にネットワークを越えるパスワードや Trivial File Transfer Protocol (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設定する場合には、`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) または特定のイネーブル 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	目的
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 [level level] {password   encryption-type encrypted-password}</code>  または <code>enable secret [level level] {password   encryption-type encrypted-password}</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。  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 文字の英数字の文字列を指定します。文字列は数字で始めることはできません。大文字と小文字が区別され、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 しか利用できません。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 Catalyst 3550 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	<code>service password-encryption</code>	(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	<code>end</code>	イネーブル EXEC モードに戻ります。
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 EXEC コマンドへのアクセスの保護

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。

特定のイネーブル 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、「[複数のイネーブル レベルの設定](#)」(p.7-9) 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。

イネーブル 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、Catalyst 3550 スイッチに物理アクセスするエンドユーザは、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は、この機能の一部をディセーブルにすることでスイッチ パスワードへのアクセスを保護します。この機能をイネーブルにした場合、エンドユーザが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せんが、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)

パスワード回復ディセーブル機能は、Catalyst 3550 ファスト イーサネット スイッチでのみ有効です。Catalyst 3550 ギガビット イーサネット スイッチでは利用できません。



(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 コピーを保存しておいてください。スイッチにはコンフィギュレーション ファイルのバックアップ コピーを保存しないでください。スイッチが VLAN Trunking Protocol (VTP; VLAN トランキング 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 コピーも保存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.38-3) 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	目的
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>no service password-recovery</code>	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 ロードがアクセスできる領域およびソフトウェア 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	<code>end</code>	イネーブル EXEC モードに戻ります。
ステップ 4	<code>show version</code>	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、手動でスイッチを起動するように設定した場合、パスワード回復のディセーブル化は機能しません。このコマンドにより、スイッチを再起動したあとでブート ロード プロンプト (`switch:`) が生成されます。

## 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	目的
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	<code>enable password <i>password</i></code>	イネーブル EXEC モードを開始します。
ステップ 3	<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 4	<code>line vty 0 15</code>	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。

## ■ イネーブル EXEC コマンドへのアクセスの保護

	コマンド	目的
ステップ 5	<code>password password</code>	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 文字の英数字の文字列を指定します。文字列は数字で始めることはできません。大文字と小文字が区別され、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	<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。  コマンド <code>line vty 0 15</code> の下にパスワードが表示されます。
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。

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、スイッチでローカルに保存します。このペアは回線またはインターフェイスに割り当てられ、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。イネーブル 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イネーブル 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	目的
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]</code> <code>{password encryption-type password}</code>	各ユーザのユーザ名、イネーブル レベル、パスワードを入力します。 <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イネーブル 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><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際に入力する必要が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。

	コマンド	目的
ステップ 4	login local	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	end	イネーブル EXEC モードに戻ります。
ステップ 6	show running-config	設定を確認します。
ステップ 7	copy running-config startup-config	(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、パスワードなしでの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。

## 複数のイネーブル レベルの設定

ソフトウェアは、デフォルトで、ユーザ EXEC とイネーブル 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。

- [コマンドのイネーブル レベルの設定 \(p.7-9\)](#)
- [回線に対するデフォルトのイネーブル レベルの変更 \(p.7-10\)](#)
- [イネーブル レベルへのログインおよび終了 \(p.7-11\)](#)

## コマンドのイネーブル レベルの設定

コマンド モードのイネーブル 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	目的
ステップ 1	configure terminal	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	privilege mode level level command	コマンドのイネーブル 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 <code>configure</code> を、EXEC モードの場合は <code>exec</code> 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 <code>interface</code> を、ライン コンフィギュレーション モードの場合は <code>line</code> 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、<code>enable</code> 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>

## ■ イネーブル EXEC コマンドへのアクセスの保護

	コマンド	目的
ステップ 3	<code>enable password level level password</code>	イネーブル 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><code>level</code> 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><code>password</code> には、1 ~ 25 文字の英数字の文字列を指定します。文字列は数字で始めることはできません。大文字と小文字が区別され、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、イネーブル 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。

コマンドをあるイネーブル 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、それぞれ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` コマンドをイネーブル レベル 14 に設定し、レベル 14 コマンドを使用する際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イネーブル レベルの変更

回線に対するデフォルトのイネーブル 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	目的
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 イネーブル レベルを変更します。  <code>level</code> 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、 <code>enable</code> 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、イネーブル 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、別のイネーブル 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イネーブル レベルを上書きできます。また、`disable` コマンドを使用すると、イネーブル レベルを低く設定できます。上位のイネーブル 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イネーブル 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イネーブル レベルを指定してください。

回線のイネーブル 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。

## イネーブル レベルへのログインおよび終了

特定のイネーブル レベルにログインし、特定のイネーブル 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	目的
ステップ 1	<code>enable level</code>	特定のイネーブル 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>	特定のイネーブル 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、アカウント管理) 機能が拡張されており、TACACS+ をイネーブルにするには AAA コマンドを使用しなければなりません。



(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 for Cisco IOS』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 \(p.7-12\)](#)
- [TACACS+ の動作 \(p.7-14\)](#)
- [TACACS+ の設定 \(p.7-14\)](#)
- [TACACS+ 設定の表示 \(p.7-19\)](#)

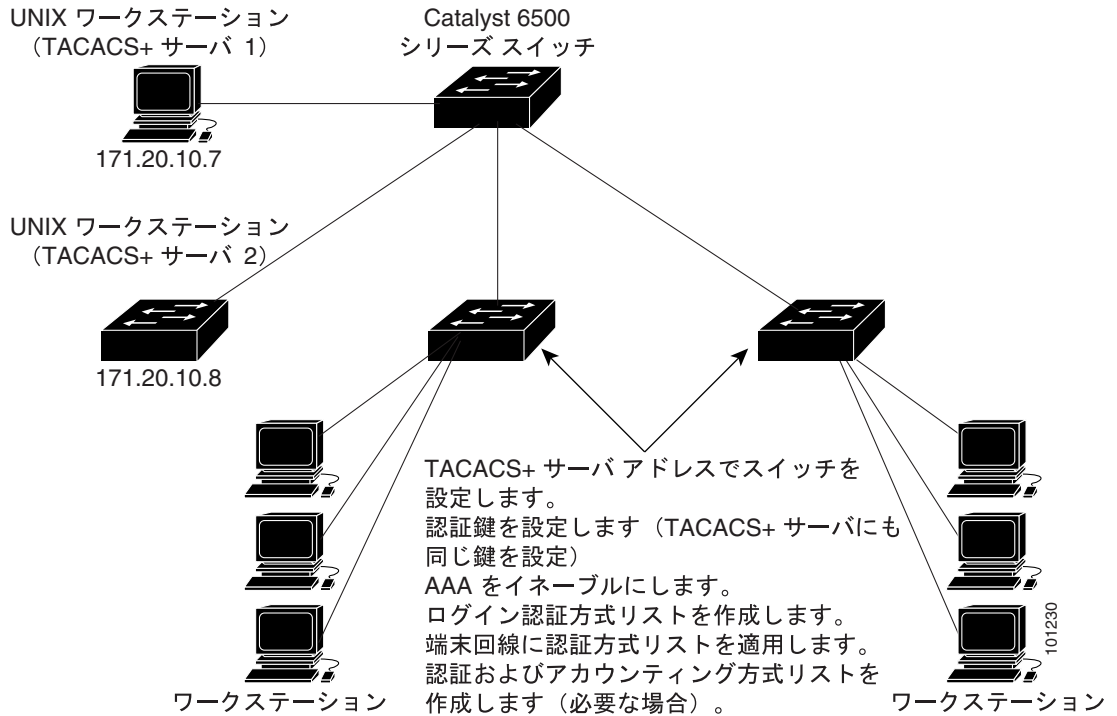
### TACACS+ の概要

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、およびアカウント管理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+ デモン) が各サービス (認証、許可、およびアカウント管理) 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可能なほか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式を提供することです。スイッチは、ほかのシスコ製ルータやアクセス 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 7-1](#) を参照 )。

図 7-1 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** ログインとパスワード ダイアログ、要求と応答、およびメッセージ 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確認する)。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。
- **アカウントिंग** 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、アカウント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できます。アカウントング 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する認証で簡易 ASCII ログインを試行すると、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+ デモンに接続してユーザ名プロンプトを取得し、ユーザに表示します。ユーザがユーザ名を入力すると、スイッチは TACACS+ デ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+ デモンに送信されます。

TACACS+ によって、デモンとユーザとの間の対話が可能になり、デ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 デモンによる認証サービスのある時点で、またはデ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、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+ デモンに接続し、デモンが 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、ユーザの認証、許可、および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の認証、許可、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 \(p.7-15\)](#)
- [TACACS+ サーバ ホストの特定と認証鍵の設定 \(p.7-15\)](#)
- [TACACS+ ログイン認証の設定 \(p.7-16\)](#)
- [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 \(p.7-18\)](#)
- [TACACS+ アカウンティングの開始 \(p.7-19\)](#)

## 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により TACACS+ を設定することはできません。イネーブルに設定されている場合、TACACS+ は CLI を使用してスイッチにアクセスするユーザを認証できます。



(注)

TACACS+ の設定は CLI を使用して行いますが、TACACS+ サーバはイネーブル レベル 15 に設定された HTTP 接続を許可します。

## TACACS+ サーバホストの特定と認証鍵の設定

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特定し、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	目的
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（複数可）を特定します。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。

## ■ TACACS+ によるスイッチ アクセスの制御

	コマンド	目的
ステップ 7	show tacacs	設定を確認します。
ステップ 8	copy running-config startup-config	(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、さまざまなインターフェイス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インターフェイスに適用してから、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インターフェイスを除いて、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ー送信を行う手順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すまで続きます。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	目的
ステップ 1	configure terminal	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	aaa new-model	AAA をイネーブルにします。

	コマンド	目的
ステップ 3	<pre>aaa authentication login {default   list-name} method1 [method2...]</pre>	<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、自動的にすべてのインターフェイスに適用されます。</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+ サーバホストの特定と認証鍵の設定</a>」(p.7-15) 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> 大文字と小文字が区別され、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	<pre>line [console   tty   vty] line-number [ending-line-number]</pre>	<p>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。</p>
ステップ 5	<pre>login authentication {default   list-name}</pre>	<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	<pre>end</pre>	<p>イネーブル EXEC モードに戻ります。</p>
ステップ 7	<pre>show running-config</pre>	<p>設定を確認します。</p>
ステップ 8	<pre>copy running-config startup-config</pre>	<p>(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。</p>

## ■ TACACS+ によるスイッチ アクセスの制御

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 を使用してログインし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	目的
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 authorization network 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ユーザ TACACS+ 許可をスイッチに設定します。
ステップ 3	<code>aaa authorization exec tacacs+</code>	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。  <code>exec</code> キーワードを指定すると、ユーザ プロファイル情報 ( <code>autocommand</code> 情報など ) が返されることがあります。
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。

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。

各イネーブル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	目的
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 を介して機能します。RADIUS をイネーブルにするには AAA コマンドを使用しなければなりません。



(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 for Cisco IOS』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 \(p.7-20\)](#)
- [RADIUS の動作 \(p.7-21\)](#)
- [RADIUS の設定 \(p.7-22\)](#)
- [RADIUS 設定の表示 \(p.7-33\)](#)

### 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対象となるシスコのルータおよびスイッチ上で稼働します。クライアントは中央 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。

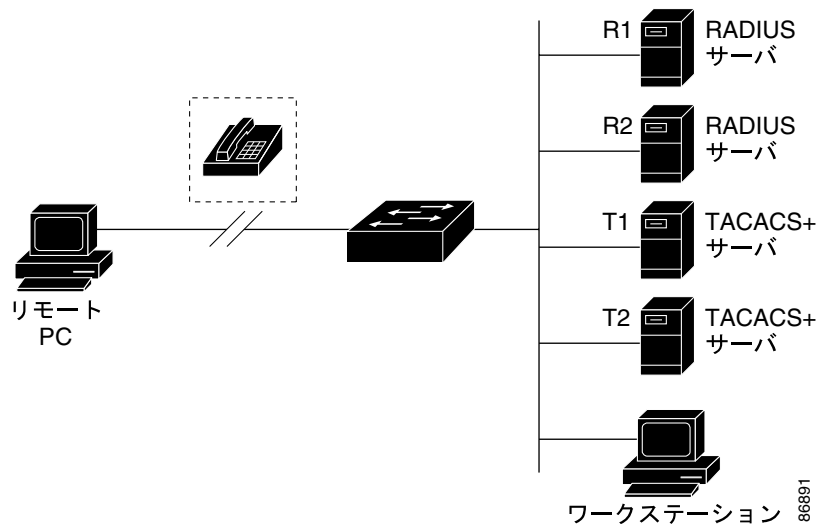
RADIUS は、アクセスのセキュリティが必要な、次のネットワーク環境で使用します。

- 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があります。あるケースでは、RADIUS を 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ます。
- 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製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 7-2 \(p.7-21\)](#) を参照してください。
- 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 8 章「IEEE 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トリング 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課金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 7-2 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みると、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 ユーザが認証されます。
  - 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 ユーザに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が稼働するホスト（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、ユーザの認証、許可、および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の認証、許可、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 \(p.7-22\)](#)
- [RADIUS サーバホストの特定 \(p.7-22\)](#) (必須)
- [RADIUS ログイン認証の設定 \(p.7-25\)](#) (必須)
- [AAA サーバグループの定義 \(p.7-27\)](#) (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 \(p.7-29\)](#) (任意)
- [RADIUS アカウントिंगの開始 \(p.7-30\)](#) (任意)
- [すべての RADIUS サーバに対する設定 \(p.7-31\)](#) (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 \(p.7-31\)](#) (任意)
- [ベンダー独自の RADIUS サーバ通信用にスイッチを設定する方法 \(p.7-33\)](#) (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 サーバホストの特定

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 ストリング
- 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。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 番目のホスト 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 デモンが稼働するホストと、そのスイッチを共用するシークレット 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の機能(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.7-31)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.7-27)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。

## ■ RADIUS によるスイッチ アクセスの制御

	コマンド	目的
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、アカウントिंग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、RADIUS サーバが応答するのを待ってスイッチが再送信するまでのインターバルを指定します。指定できる範囲は 1 ~ 1000 秒です。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が設定されていない場合は、<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 回です。<code>radius-server host</code> コマンドで再送信の値が設定されない場合は、<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のあるテキスト スtring 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 アドレスに関連付けられた複数のホスト 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リ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 テキスト 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、さまざまなインターフェイス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インターフェイスに適用してから、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インターフェイスを除いて、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手順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すまで続きます。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	目的
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	目的
ステップ 3	aaa authentication login {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 <i>指定されていない</i> 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、自動的にすべてのインターフェイスに適用されます。</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 サーバホストの特定</a>」(p.7-22)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> 大文字と小文字が区別され、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	line [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	login authentication {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	end	イネーブル EXEC モードに戻ります。
ステップ 7	show running-config	設定を確認します。
ステップ 8	copy running-config startup-config	(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、各エントリが一意の識別子 (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に同じサービス (たとえば、アカウントिंग) を設定すると、設定された 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。

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。

## ■ RADIUS によるスイッチ アクセスの制御

	コマンド	目的
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、RADIUS サーバが応答するのを待ってスイッチが再送信するまでのインターバルを指定します。指定できる範囲は 1 ~ 1000 秒です。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が設定されていない場合は、<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 回です。<code>radius-server host</code> コマンドで再送信の値が設定されない場合は、<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> <b>(注)</b> 鍵は、RADIUS サーバ上で使用する暗号化鍵と照合する必要の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 アドレスに関連付けられた複数のホスト 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	目的
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.7-25) 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。

```
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 を使用してログインし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 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。

## ■ RADIUS によるスイッチ アクセスの制御

	コマンド	目的
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 authorization network radius</code>	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	<code>aaa authorization exec radius</code>	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。  exec キーワードを指定すると、ユーザ プロファイル情報 ( <code>autocommand</code> 情報など ) が返されることがあります。
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。

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 イネーブル 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	目的
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 radius</code>	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	<code>aaa accounting exec start-stop radius</code>	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、イネーブル EXEC プロセスの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。

## 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コンフィギュレーションを設定するには、イネーブル EXEC モードで次の手順を実行します。

	コマンド	目的
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 回で、指定できる範囲は 1 ~ 1000 回です。
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 秒です。
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、および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。

## ■ RADIUS によるスイッチ アクセスの制御

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"
```

次の例は、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方法を示します。

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

次の例は、RADIUS サーバ データベース内の許可 VLAN を指定する方法を示しています。

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

次の例は、この接続中に ASCII 形式の入力 Access Control List (ACL; アクセス制御リスト) をインターフェイスに適用する方法を示しています。

```
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 decnet-iv"
```

次の例は、この接続中に ASCII 形式の出力 ACL をインターフェイスに適用する方法を示しています。

```
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、独自に一意的ベンダー 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	目的
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) <code>accounting</code> キーワードを使用して、認識される VSA 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) <code>authentication</code> キーワードを使用して、認識される VSA 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 VSA 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 VSA 26 の詳細については、『Cisco IOS Security Configuration Guide for Cisco IOS』Release 12.2 の付録「RADIUS Attributes」を参照してください。

## ベンダー独自の RADIUS サーバ通信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独自情報の通信方式を規定していますが、一部のベンダーは、独自の方法で RADIUS アトリビュートを機能拡張しています。Cisco IOS ソフトウェアは、ベンダー独自仕様の 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 (ベンダー独自または IETF のドラフト準拠) を設定するには、RADIUS サーバ デ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tring 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tring 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。

ベンダー独自の RADIUS サーバ ホスト、および共有シークレット テキスト スtring 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	目的
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} non-standard</code>	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、ベンダー独自の実装 RADIUS を使用していることを明確にします。
ステップ 3	<code>radius-server key string</code>	<p>スイッチとベンダー独自の 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tring 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tring 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のあるテキスト スtring 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	<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。

ベンダー独自の 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。鍵を削除するには、`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、ベンダー独自の 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、スイッチに暗号化マルチレイヤ 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 からこの暗号化ソフトウェア ファイルをダウンロード するには、許可を得る必要があります。詳細については、この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 \(p.7-34\)](#)
- [Kerberos の動作 \(p.7-36\)](#)
- [Kerberos の設定 \(p.7-38\)](#)

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、信頼のおけるサードパーティとして Catalyst 3550 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することを主な目的としています。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 サーバとして設定可能で Kerberos プロトコルでユーザを認証できる Catalyst 3550 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（ほか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)ほかのネットワーク 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
- rsh (リモート シェル プロトコル)

表 7-2 に、一般的な Kerberos 関連用語とその定義を示します。

表 7-2 Kerberos の用語





用語	定義
認証	ユーザやサービスがほかのサービスに対して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、スイッチはほか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所有していて、またどのような動作を実行できるかを、スイッチが決定する手段。
証明書	認証チケット (TGT <sup>1</sup> や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	<p>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。</p> <p> (注) Kerberos プリンシパル名およびインスタンス名はすべて小文字でなければなりません。</p> <p> (注) Kerberos レルム名はすべて大文字でなければなりません。</p>
KDC <sup>2</sup>	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鍵発行局。
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。

表 7-2 Kerberos の用語（続き）

用語	定義
Kerberos レルム	<p>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。</p> <p> (注) Kerberos レルム名はすべて大文字でなければなりません。</p>
Kerberos サーバ	<p>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 サーバにクエリーを送信して、ほかのネットワーク サービスの認証を得ます。</p>
KEYTAB <sup>3</sup>	<p>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 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 以前のバージョンでは、KEYTAB は SRVTAB<sup>4</sup> と呼びます。</p>
プリンシパル	<p>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。</p> <p> (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。</p>
サービス証明書	<p>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。</p>
SRVTAB	<p>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。</p>
TGT	<p>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 が表した Kerberos レルム内のネットワーク サービスに対して認証を得ることができます。</p>

1. TGT = Ticket Granting Ticket (身分証明書)
2. KDC = Key Distribution Center (鍵発行局)
3. KEYTAB = Key Table (キー テーブル)
4. SRVTAB = Server Table (サーバ テーブル)

## Kerberos の動作

ここでは、ネットワーク セキュリティ サーバとして設定されたスイッチでの Kerberos の動作方法について説明します。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 サーバとしてのスイッチを使用してネットワーク 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 \(p.7-37\)](#)
2. [KDC からの TGT の取得 \(p.7-37\)](#)
3. [ネットワーク サービスに対する認証の取得 \(p.7-37\)](#)



(注)

Kerberos サーバには、ネットワーク セキュリティ サーバとして設定可能で Kerberos プロトコルを使用してリモート ユーザを認証できるスイッチを使用できます。

## 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。リモート ユーザが境界スイッチに対して認証を得る際、次の処理が発生します。

1. ユーザが境界スイッチに対して、Kerberos 未対応の Telnet 接続を開始します。
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 KDC からの TGT を要求します。
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場合、ユーザはスイッチに対して認証を得ます。
  - 暗号解除に失敗した場合、ユーザはユーザ名およびパスワード (Caps Lock または Num Lock のオン / オフに注意) を再入力するか、または別のユーザ名およびパスワードを入力することで、ステップ 2 を繰り返します。

Kerberos 未対応の Telnet セッションを開始し、境界スイッチの認証を得ているリモート 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 サービスにアクセスする前に KDC から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用できないからです。

## 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 番目のセキュリティ レイヤについて説明します。ユーザは、ネットワーク サービス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 サービス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 番目のセキュリティ 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相互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際の注意事項は次のとおりです。

- 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 サーバとして設定可能で Kerberos プロトコルを使用してユーザを認証できる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。

設定手順については、『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	目的
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。 <code>default</code> 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インターフェイスに適用されます。
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 データベースを確認してそのユーザに EXEC シェルの実行を許可するかどうかを判別します。
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ユーザ 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><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イネーブル 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><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際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 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。SSH は、制限をエクスポートする暗号化セキュリティ機能です。この機能を使用するには、暗号化された IP サービス イメージ (以前は Enhanced Multilayer Software Image [ EMI; 拡張マルチレイヤ ソフトウェア イメージ ]) をスイッチに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 からこの暗号化ソフトウェア ファイルをダウンロードするには、許可を得る必要があります。詳細については、この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.7-40 )
- SSH の設定 ( p.7-41 )
- SSH 設定およびステータスの表示 ( p.7-43 )

SSH の設定例については、『Cisco IOS Security Configuration Guide』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、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 はデバイスに対する安全なリモート接続を可能に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 リリースでは、SSH バージョン 1( SSHv1 )および SSH バージョン 2( SSHv2 ) をサポートしています。

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、およびサポートされるバージョン ( p.7-40 )
- 制限事項 ( p.7-41 )

### SSH サーバ、統合クライアント、およびサポートされるバージョン

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 クライアントを使用して、SSH サーバを稼働しているスイッチに接続できます。SSH サーバは、このリリースでサポートされている SSH クライアントおよび他社製の SSH クライアントで機能します。SSH クライアントは、このリリースでサポートされている SSH サーバおよび他社製の SSH サーバでも機能します。

スイッチは、SSHv1 または SSHv2 サーバをサポートします。

スイッチは、SSHv1 クライアントをサポートします。

SSH は、DES という暗号化アルゴリズム、Triple DES ( 3DES ) 暗号化アルゴリズム、および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.7-12] 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.7-20] 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.7-39] ) を参照 )



(注)

このソフトウェアリリースは、IP Security ( IPsec ) をサポートしません。

## 制限事項

SSH には次の制限があります。

- スイッチは、Rivest, Shamir, and Adelman ( RSA ) 認証をサポートします。
- SSH がサポートするのは、シェルで実行するアプリケーションのみです。
- 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 ) 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.7-41 )
- [SSH を稼働させるためのスイッチの設定](#) ( p.7-42 ) ( 必須 )
- [SSH サーバの設定](#) ( p.7-43 ) ( SSH サーバとしてスイッチを設定する場合のみ、必須 )

## 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 サーバによって使用できます。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に CLI エラー メッセージを受信した場合、RSA 鍵ペアは生成されません。ホスト名およびドメインを再設定してから、`crypto key generate rsa` コマンドを入力します。詳細については、「[SSH を稼働させるためのスイッチの設定](#)」( p.7-42 ) を参照してください。
- RSA 鍵ペアを生成するとき、[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。このメッセージが表示された場合、`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、[No domain specified] というメッセージが表示される場合があります。メッセージが表示された場合、`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。
- ローカル認証および許可認証方式を設定するとき、コンソール上では AAA がディセーブルであることを確認してください。

## SSH を稼働させるためのスイッチの設定

SSH を稼働するようスイッチを設定するには、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。このステップは必須です。詳細については、このリリースのリリース ノートを参照してください。
2. ホスト名および IP ドメイン名をスイッチに設定します。SSH サーバとしてスイッチを設定する場合のみ、この手順を実行します。
3. スイッチに RSA 鍵ペアを生成します。このスイッチは自動的に SSH をイネーブルにします。SSH サーバとしてスイッチを設定する場合のみ、この手順を実行します。
4. ローカルまたはリモート アクセスへの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.7-39) 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、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。SSH サーバとしてスイッチを設定する場合のみ、この手順は必須です。

	コマンド	目的
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 ドメイン名を設定します。
ステップ 4	<code>crypto key generate rsa</code>	スイッチでのローカルおよびリモート認証のため、SSH サーバをイネーブルにし、RSA 鍵ペア を生成します。  最小係数を 1024 ビットにすることを推奨します。  RSA 鍵を生成すると、係数長を入力するよう指示されます。係数長が長いと安全性が高くなりますが、生成および使用する時間がより長くなります。
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。  スイッチで 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 を削除したあと、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	目的
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>	(任意) SSH バージョン 1 または SSH バージョン 2 を実行するようスイッチを設定します。 <ul style="list-style-type: none"> <li>1 SSH バージョン 1 を実行するようスイッチを設定します。</li> <li>2 SSH バージョン 2 を実行するようスイッチを設定します。</li> </ul> このコマンドを入力しない、またはキーワードを指定しない場合、SSH サーバは SSH クライアントでサポートされる最新の SSH バージョンを選択します。たとえば、SSH クライアントが SSHv1 および SSHv2 をサポートする場合、SSH サーバは SSHv2 を選択します。
ステップ 3	<code>ip ssh {timeout seconds   authentication-retries number}</code>	SSH コンソール パラメータを設定します。 <ul style="list-style-type: none"> <li>タイムアウト値を秒単位で指定します。デフォルト値は 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 CLI ベースのセッションのデフォルト タイムアウト値を使用します。</li> </ul> デフォルトでは、ネットワークでの CLI ベース マルチ セッションに対応した、最大 5 つの同時暗号化 SSH 接続を利用できます (セッション 0 ~ 4)。シェル実行を開始すると、CLI ベース セッション タイムアウト値は 10 分のデフォルトに戻ります。 <ul style="list-style-type: none"> <li>クライアントがサーバに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> 両方のパラメータを設定する場合、このステップを繰り返します。
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 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。  スイッチで SSH サーバ接続のステータスを表示します。
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。

デフォルト SSH 制御パラメータに戻すには、`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 設定およびステータスの表示

SSH サーバ設定およびステータスを表示するには、表 7-3 に示すイネーブル EXEC コマンドの 1 つまたは複数を使用します。

表 7-3 SSH サーバ設定およびステータスを表示するコマンド

コマンド	目的
<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。

これらのコマンドについては、『Cisco IOS Security Command Reference』Release 12.2 の「Other Security Features」の章にある「Secure Shell 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/ios122/122cgr/fsecr\\_r/fothercr/srfssh.htm](http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgr/fsecr_r/fothercr/srfssh.htm)

## SSL HTTP のためのスイッチの設定

ここでは、HTTP1.1 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、およびメッセージ完全性、および HTTP クライアント認証を提供し、安全な HTTP 通信が可能になります。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 からこの暗号化ソフトウェア ファイルをダウンロードするには、許可を得る必要があります。暗号化イメージの詳細については、この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 \(p.7-44\)](#)
- [セキュア HTTP サーバおよびクライアントの設定 \(p.7-46\)](#)
- [セキュア HTTP サーバおよびクライアント ステータスの表示 \(p.7-50\)](#)

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

安全な 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に送られる前に暗号化されます。SSL 暗号化を使用した HTTP では安全な接続を提供していて、Web ブラウザからそのような機能をスイッチに設定できます。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 の実装を使用します。HTTP over SSL は HTTPS と略されます。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 サーバに渡して、その結果、元の要求に応答することになります。

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 ユーザ エージェントサービスを実行し、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 Authorities (CA; 認証局) は、認証要求を管理し、参加しているネットワーク デバイスに証明書を発行します。これらのサービスは、参加するデバイスに中央集中型のセキュリティ鍵と証明書管理を提供します。特定の CA サーバをトラストポイントと呼びます。

接続試行が実行されると、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.509v3 証明書をクライアントに発行することで、安全な接続が可能になります。その結果、クライアント（通常は Web ブラウザ）が証明書を認証できる公開鍵を持つことができます。

安全な HTTP 接続のために、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、サーバは自ら認証し必要な 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、接続クライアントは、証明書が自己認証されたものであり、ユーザは接続を許可することも拒否することもできる旨の通知を生成します。このオプションは、内部ネットワーク ポロジ（テスト等）で便利です。

CA トラストポイントを設定しない場合、安全な 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）に対して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消失し、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のままなので、次のセキュア HTTP 接続をイネーブルにした場合にも使用できます。

自己署名証明書が生成されたら、この情報は `show running-config` イネーブル EXEC コマンドの出力に含まれます。これは、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です。

```
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)

この自己署名証明書は、セキュア HTTP サーバをディセーブルにして `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ことで削除できます。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ます。



(注)

`TP self-signed` 以降の値は、デバイスのシリアル番号により異なります。

オプションのコマンド (`ip http secure-client-auth`) を使用して、HTTPS サーバが X.509v3 証明書をクライアントから要求できます。クライアントの認証には、サーバ自身によるサーバ認証よりもセキュリティが確保されています。

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

## CipherSuite

CipherSuite 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 アルゴリズムを指定します。HTTPS サーバに接続するときに、クライアント Web ブラウザはサポート済みの CipherSuite リストを提供し、クライアントとサーバはこのリストの中から両方でサポートされている最良の暗号化アルゴリズムをネゴシエーションし、使用します。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、および DES-EDE3-CBC をサポートしています。

可能なかぎり最善の暗号化方式を使用できるようにするためには、Microsoft Internet Explorer バージョン 5.5 以降または Netscape Communicator バージョン 4.76 以降などの、128 ビット暗号化方式を使用できるクライアント 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 ビット暗号化方式を使用しないため、他の CipherSuite よりもセキュリティの面で劣ります。

CipherSuite がより安全で複雑になるほど、処理時間が多少必要になります。このリストは、スイッチでサポートされる CipherSuite を定義するもので、ルータ処理負荷（速度）の面で早さの順にランク付けされています。

1. メッセージ暗号化用に DES-CBC と、メッセージ ダイジェスト用に SHA を使用した、SSL\_RSA\_WITH\_DES\_CBC\_SHA RSA 鍵交換（RSA 公開鍵暗号化方式）
2. RC4 128 ビット暗号化方式を使用しメッセージ ダイジェスト用に MD5 を使用した SSL\_RSA\_WITH\_RC4\_128\_MD5 RSA 鍵交換
3. RC4 128 ビット暗号化方式を使用しメッセージ ダイジェスト用に SHA を使用した SSL\_RSA\_WITH\_RC4\_128\_SHA RSA 鍵交換
4. メッセージ暗号化用に 3DES と DES-EDE3-CBC、メッセージ ダイジェスト用に SHA を使用した、SSL\_RSA\_WITH\_DES\_CBC\_SHA RSA 鍵交換（RSA 公開鍵暗号化方式）

（指定された暗号化方式とダイジェスト アルゴリズムの組み合わせとともに）RSA は、双方の SSL 接続の鍵生成および認証に使用されます。この RSA の使用状況は、CA トラストポイントが設定されているかどうかには関係ありません。

## 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上に SSL を設定する手順について説明します。その手順は次のとおりです。

- [SSL のデフォルト設定 \(p.7-46\)](#)
- [SSL 設定時の注意事項 \(p.7-47\)](#)
- [CA トラストポイントの設定 \(p.7-47\)](#)
- [セキュア HTTP サーバの設定 \(p.7-48\)](#)
- [セキュア HTTP クライアントの設定 \(p.7-49\)](#)

## 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

SSL がスイッチ クラスタで使用される場合、SSL セッションがクラスタ コマンドで終了します。クラスタ メンバー スイッチで標準 HTTP を実行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、日付が間違っている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

安全な HTTP 接続のために、正式な 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行います。


	コマンド	目的
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 (今までホスト名を指定していない場合のみ必要)。ホスト名はセキュリティ 鍵と証明書に必要です。
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 (今まで IP ドメイン名を指定していない場合のみ必要)。ドメイン名はセキュリティ 鍵と証明書に必要です。
ステップ 4	<code>crypto key generate rsa</code>	(任意) RSA 鍵ペアを作成します。RSA 鍵ペアは、スイッチから認証を取得する前に必要です。RSA 鍵ペアが自動的に生成されます。必要に応じて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 コンフィギュレーション 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 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 サーバを介して CA から証明書を取得するようにスイッチを設定します。
ステップ 8	<code>crl query url</code>	ピアの証明書が失効していないことを確認するために、Certificate Revocation List (CRL; 証明書失効リスト) を要求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) トラストポイントが 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使用され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 コンフィギュレーション モードを終了してグローバル 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 で使用した同じ名前を使用します。
ステップ 12	<code>crypto ca enroll name</code>	指定した CA とらストポイントから証明書を取得します。このコマンドは各 RSA 鍵ペアに対して署名済みの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 に関連したすべての ID 情報と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

認証用に認証局を使用している場合、HTTP サーバをイネーブルにする前に前述の手順を使用してスイッチに CA トラストポイントを設定します。CA トラストポイントを設定していない場合、セキュア HTTP サーバを最初にイネーブルにするときに自己署名証明書が生成されます。サーバの設定後、標準およびセキュア 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、最大接続数、タイムアウト ポリシー）を任意で設定できます。

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	目的
ステップ 1	<code>show ip http server status</code>	(任意) 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を判断するために、HTTP サーバのステータスを表示します。出力は次のいずれかの行が表示されます。  HTTP secure server capability: Present or 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の場合、これをイネーブルにします。HTTPS サーバはデフォルトでイネーブルです。
ステップ 4	<code>ip http secure-port port-number</code>	(任意) HTTPS サーバに使用されるポート番号を指定します。デフォルトのポート番号は 433 です。有効なオプションは 433 か、または 1025 ~ 65535 の範囲の番号です。
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 接続の暗号化方式で使用する 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ネゴシエーションして双方がサポートする CipherSuite を使用できるようにします。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、認証用にクライアントから X.509v3 証明書を要求するように HTTPS 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はしません。
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ために CA トラストポイントを使用するように指定します。   (注) このコマンドを使用すると、前の手順に沿ってすでに CA トラストポイントが設定されているとみなされます。
ステップ 8	<code>ip http path path-name</code>	(任意) HTML ファイル用の HTTP パスを設定します。パスは、ローカルシステム上の HTTP サーバファイルの位置を指定します(通常はシステムフラッシュメモリにあります)。
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可能にするために使用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 サーバに許可された最大同時接続数を設定します。指定できる範囲は 1 ~ 16 で、デフォルトは 5 です。

	コマンド	目的
ステップ 11	<code>ip http timeout-policy idle seconds life seconds requests value</code>	(任意) 定義された状況下で HTTP サーバへの接続がオープンのまま でいる時間を設定します。  <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大 時間間隔。指定できる範囲は 1 ~ 600 秒です。デフォルトは 180 秒 (3 分) です。</li> <li><b>life</b> 接続が確立されている最大時間間隔。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> 持続的な接続で処理される最大要求数。最大値は 86400 で、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>	設定を確認するために HTTP セキュア サーバのステータスを表示し 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るには、`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件を削除するには、`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` を入力します。ここで 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ている場合、URL の後ろにポート番号も入力する必要があります。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。認証局はセキュア HTTP クライアントの証明書が必要です。この手順では、すでに CA トラストポイントをスイッチに設定しているものとみなしています。CA トラストポイントが未設定でリモート HTTPS サーバがクライアント認証を必要とする場合、セキュア HTTP クライアントへの接続が失敗します。

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	目的
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する場合に 使用される CA トラストポイントを指定します。このコマンドを使用 すると、前の手順を使用してすでに CA トラストポイントが設定 されているとみなされます。クライアント認証が不要な場合または 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 は任意です。

## ■ SSL HTTP のためのスイッチの設定

	コマンド	目的
ステップ 3	<code>ip http client secure-ciphersuite</code> {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}	(任意)HTTPS 接続の暗号化方式で使用する 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ネゴシエーションして双方がサポートする CipherSuite を使用できるようにします。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>	設定を確認するために HTTP セキュア サーバのステータスを表示します。
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。

クライアント 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。すでに設定されているクライアントの CipherSuite 仕様を削除するには、`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 ステータスの表示

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 7-4 のイネーブル EXEC コマンドを使用します。

表 7-4 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	目的
<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 接続の生成された自己署名証明書を表示します。

## Secure Copy Protocol のためのスイッチの設定

Secure Copy (SCP; セキュア コピー) 機能は、スイッチ コンフィギュレーション ファイルまたはスイッチ イメージ ファイルをコピーする安全な認証方法です。SCP は SSH (Berkeley r-tool の代わりにセキュリティの高い代替品を提供するアプリケーションおよびプロトコル) を利用します。

SSH を動作させるには、スイッチは RSA 公開鍵と秘密鍵のペアが必要です。これは SCP も同様で、安全な転送のために SSH を使用します。

SSH も AAA 認証 を利用し、さらに SCP も AAA 認証 を利用しているので、正しく設定する必要があります。

- SCP をイネーブルにする前に、スイッチに SSH、認証、許可を正しく設定する必要があります。
- SCP は安全な転送のために SSH を使用するので、スイッチは RSA 鍵ペアを持つ必要があります。



(注)

SCP を使用する場合、copy コマンドにパスワードを入力できません。パスワードは要求されたときに入力する必要があります。

SCP 機能を設定するには、以下の概念を理解する必要があります。

- SCP の動作は、SCP がセキュリティに SSH を使用することを除き、Berkeley r-tool スイートのリモート コピー (rcp) の機能と似ています。また SCP は、AAA 認証を設定する必要があるため、ルータはユーザが正しいイネーブル レベルを有しているかどうかを判断できます。
- 適切な認証を得ているユーザは SCP を使用して copy コマンドで Cisco IOS File System (IFS) 内の任意のファイルをスイッチにコピーできます。認証済みの管理者は、この作業をワークステーションからも実行できます。

SCP の設定および確認方法については、『Cisco IOS New Features, Cisco IOS』Release 12.2 の「Secure Copy Protocol」の章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/122t2/ftscp.htm>

